绿盾日志分析

目录

[1 透明加解密 9](#_Toc49522335)

[1.1 透明加解密——配置日志 9](#_Toc49522336)

[1.1.1 受控进程的配置 9](#_Toc49522337)

[1.1.2 EncrytType行 9](#_Toc49522338)

[1.1.2.1 注：EncryptType行CtrlCode值 10](#_Toc49522339)

[1.1.3 密级 12](#_Toc49522340)

[1.1.4 特殊目录 12](#_Toc49522341)

[1.1.5 特殊文件后缀 13](#_Toc49522342)

[1.1.6 受控程序高级设置 13](#_Toc49522343)

[1.2 透明加解密——现象日志 14](#_Toc49522344)

[1.2.1 非受控程序打开加密文件 14](#_Toc49522345)

[1.2.2 受控程序打开加密文件 14](#_Toc49522346)

[1.2.3 写加密头 15](#_Toc49522347)

[1.2.4 常用文件后缀加密 16](#_Toc49522348)

[1.2.5 密级不够无权打开加密文件 17](#_Toc49522349)

[1.3 落地加密 18](#_Toc49522350)

[1.3.1 落地加密——配置日志 18](#_Toc49522351)

[1.3.2 落地加密——现象日志 19](#_Toc49522352)

[1.3.2.1附：驱动上报应用层加解密的编号列表 19](#_Toc49522353)

[1.3.3 落地加密——查看例外 19](#_Toc49522354)

[1.4 半透明加密 20](#_Toc49522355)

[1.4.1 半透明加密——配置日志 20](#_Toc49522356)

[1.4.2 半透明加密——现象日志 21](#_Toc49522357)

[1.5 智能加密 22](#_Toc49522358)

[1.5.1 智能加密——配置日志 22](#_Toc49522359)

[1.5.1.1 ArWorstation.log 22](#_Toc49522360)

[1.5.1.2 ArDriver.log 22](#_Toc49522361)

[1.5.1.3 ContentAware.log（具体敏感策略） 22](#_Toc49522362)

[1.5.1.3.1 附-常规正则表达式 23](#_Toc49522363)

[1.5.2 敏感策略案例分析 24](#_Toc49522364)

[1.5.2.1 关键字规则 24](#_Toc49522365)

[1.5.2.2 正则表达式规则 25](#_Toc49522366)

[1.5.3 智能加密——现象日志 25](#_Toc49522367)

[1.5.3.1 ArDriver.log 25](#_Toc49522368)

[1.5.3.2 Arworstation.log 25](#_Toc49522369)

[1.5.3.3 ContentAware.log（接受上报，进行分析文件内容） 26](#_Toc49522370)

[1.5.3.4 ArWorkstation.log(服务端接收ldcontentaware结果后的加解密结果) 26](#_Toc49522371)

[1.6 应用层加解密 27](#_Toc49522372)

[1.6.1全盘加密 27](#_Toc49522373)

[1.6.2 全盘解密 28](#_Toc49522374)

[1.6.3 批量加解密 29](#_Toc49522375)

[1.6.3.1 批量加密失败代码 29](#_Toc49522376)

[1.6.3.2 批量解密失败代码 30](#_Toc49522377)

[1.6.3.3 关于批量加解密日志查看方法 30](#_Toc49522378)

[1.6.4 驱动上报应用层加解密 31](#_Toc49522379)

[1.7 压缩包穿透加解密日志 33](#_Toc49522380)

[1.7.1 压缩包穿透加解密——配置日志 33](#_Toc49522381)

[1.7.2 压缩包穿透加解密——现象日志 34](#_Toc49522382)

[1.7.3 压缩包穿透加解密——错误码 35](#_Toc49522383)

[2 CoreConfig配置日志 35](#_Toc49522384)

[3 升级加密文件 36](#_Toc49522385)

[3.1 升级成功示例 38](#_Toc49522386)

[3.2 升级失败示例 38](#_Toc49522387)

[3.3 附:升级加密文件错误码列表（同压缩包穿透加解密和批量加解密错误码） 39](#_Toc49522388)

[4 Linux日志 41](#_Toc49522389)

[4.1 tr\_service.log 41](#_Toc49522390)

[4.1.1 Linux加密客户端版本 41](#_Toc49522391)

[4.1.2 用户登录 41](#_Toc49522392)

[4.1.3 客户端与服务器链接状态 41](#_Toc49522393)

[4.1.4 策略更新 42](#_Toc49522394)

[4.2 trld.log 44](#_Toc49522395)

[4.3 获取受控程序md5值 45](#_Toc49522396)

[5 macOS日志 46](#_Toc49522397)

[5.1 trservice.log 46](#_Toc49522398)

[5.1.1 没有分配加密模块，继而驱动停止工作 47](#_Toc49522399)

[5.1.2 导入离线离线策略失败 47](#_Toc49522400)

[5.2 trld.log 49](#_Toc49522401)

[6 审批日志 50](#_Toc49522402)

[6.1 集成审批日志 50](#_Toc49522403)

[6.1.1 申请解密失败错误码 52](#_Toc49522404)

[6.2 独立审批日志（待补充） 53](#_Toc49522405)

[6.3 邮件白名单日志 54](#_Toc49522406)

[6.3.1邮件白名单——配置日志 54](#_Toc49522407)

[6.3.1.1 查看发送加密附件给收件人白名单情况 54](#_Toc49522408)

[6.3.1.1.1 收件人是白名单情况 54](#_Toc49522409)

[6.3.1.1.2 收件人是非白名单情况 55](#_Toc49522410)

[6.3.1.1.3 收件人既有白名单也有非白名单用户 56](#_Toc49522411)

[6.3.2 发件白名单发送加密附件情况 59](#_Toc49522412)

[6.4 其他日志关键字 60](#_Toc49522413)

[7 服务器白名单日志 60](#_Toc49522414)

[7.1 旧服务器白名单 60](#_Toc49522415)

[7.1.1 查看驱动是否正常 60](#_Toc49522416)

[7.1.2 确认服务器白名单策略信息是否下发 61](#_Toc49522417)

[7.1.3 确认是否有转链信息 61](#_Toc49522418)

[7.1.4 确认是否有解析到上传文件 61](#_Toc49522419)

[7.1.5 确认是否进入解密流程 62](#_Toc49522420)

[7.1.6 确认是否有解析到下载文件 62](#_Toc49522421)

[7.1.7 确认是否进入加密流程 62](#_Toc49522422)

[7.2 新服务器白名单 63](#_Toc49522423)

[7.2.1 LdFileGate.log 63](#_Toc49522424)

[7.2.2 LdNetMon.log 63](#_Toc49522425)

[7.2.2.1 访问白名单网站 64](#_Toc49522426)

[7.2.2.2 开着白名单网站的浏览器访问非白名单网站被禁止 65](#_Toc49522427)

[7.2.3 ArDriver.log查看白名单设置和加解密过程 65](#_Toc49522428)

[7.2.3.1 附：驱动上报应用层加解密的编号列表 66](#_Toc49522429)

[8 应用服务器安全接入系统日志 66](#_Toc49522430)

[8.1 应用服务器安全接入系统日志——配置日志 66](#_Toc49522431)

[8.2 应用服务器安全接入系统日志——现象日志 67](#_Toc49522432)

[8.2.1 正常转链日志 67](#_Toc49522433)

[8.2.2 UDP错误日志 67](#_Toc49522434)

[8.2.3 TCP错误日志 67](#_Toc49522435)

[8.2.4 错误8199日志 68](#_Toc49522436)

[8.3 附：接入系统错误码 68](#_Toc49522437)

[8.4 应用接入系统macOS日志说明 69](#_Toc49522438)

[8.4.1 查看代理是否运行 69](#_Toc49522439)

[8.4.2 关键日志说明 69](#_Toc49522440)

[8.4.3 部分问题排查说明 70](#_Toc49522441)

[8.4.3.1 查看具体的转链信息 70](#_Toc49522442)

[8.4.3.2 查看访问的数据是否有经过代理服务器转发 70](#_Toc49522443)

[8.4.4 应用接入系统Linux日志说明 71](#_Toc49522444)

[8.4.4.1 查看代理是否运行、重启代理的方法 71](#_Toc49522445)

[8.4.4.2 关键日志说明 71](#_Toc49522446)

[8.4.4.3部分问题排查说明 72](#_Toc49522447)

[9 离线策略日志 73](#_Toc49522448)

[9.1 离线策略日志--现象日志 74](#_Toc49522449)

[9.1.1 提示“装载离线策略文件失败：时间定义非法” 74](#_Toc49522450)

[9.1.1 提示“装载离线策略文件失败:该文件不允许被导入” 74](#_Toc49522451)

[10 水印设置日志 74](#_Toc49522452)

[10.1 屏幕水印设置 74](#_Toc49522453)

[10.2 打印水印设置日志 75](#_Toc49522454)

[10.3 外发水印设置日志 75](#_Toc49522455)

[10.3.1 外发屏幕水印 75](#_Toc49522456)

[10.3.2 外发打印水印 76](#_Toc49522457)

[10.4 word文档水印设置日志 77](#_Toc49522458)

[11 上网限制日志 77](#_Toc49522459)

[11.1 上网限制日志——配置日志 77](#_Toc49522460)

[11.1.2 端口限制 77](#_Toc49522461)

[11.1.3 网站限制 77](#_Toc49522462)

[11.2 上网限制日志——现象日志 78](#_Toc49522463)

[12 应用程序限制日志 79](#_Toc49522464)

[12.1 应用程序限制日志——配置日志 79](#_Toc49522465)

[12.1.1 应用程序限制 79](#_Toc49522466)

[12.1.2 窗口标题限制 79](#_Toc49522467)

[12.1.3 禁止受控程序改名 80](#_Toc49522468)

[12.2 应用程序限制日志——现象日志 81](#_Toc49522469)

[12.2.1 应用程序限制 81](#_Toc49522470)

[12.2.2 窗口标题限制 82](#_Toc49522471)

[12.2.3 禁止受控程序改名 82](#_Toc49522472)

[13 USB存储设备限制日志 83](#_Toc49522473)

[13.1 USB存储设备限制日志——配置日志 83](#_Toc49522474)

[13.1.1 USB存储设备限制 83](#_Toc49522475)

[13.1.2 USB存储设备认证 83](#_Toc49522476)

[13.2 USB存储设备限制日志——现象日志 83](#_Toc49522477)

[13.2.1 插入U盘后，看USB策略 83](#_Toc49522478)

[13.2.2 插入USB存储设备后，看设备ID 84](#_Toc49522479)

[14 行为审计系统 84](#_Toc49522480)

[14.1 文件监视过滤配置 84](#_Toc49522481)

[14.2 驱动日志 85](#_Toc49522482)

[15 其他日志（绿盾版本、分区名、工作模式、模块编号、MTP等） 86](#_Toc49522483)

[15.1 绿盾版本 86](#_Toc49522484)

[15.2 网卡MAC 86](#_Toc49522485)

[15.3 分区名 86](#_Toc49522486)

[15.4 时间规则 87](#_Toc49522487)

[15.5 工作模式 88](#_Toc49522488)

[15.5.1 指令策略更新 88](#_Toc49522489)

[15.5.2 策略生效 88](#_Toc49522490)

[15.5.2.1 工作模式启用加密 88](#_Toc49522491)

[15.5.2.2 工作模式关闭加密 88](#_Toc49522492)

[15.5.2.3 附：驱动停止工作代码byCantWorkCause 89](#_Toc49522493)

[15.6 查看功能模块是否启用 89](#_Toc49522494)

[15.6.1 模块编号列表 89](#_Toc49522495)

[15.7 操作员名称、操作员编号、分组编号、文件密级等 91](#_Toc49522496)

[15.7.1 附：默认的文件密级对应关系 91](#_Toc49522497)

[15.7.2 允许或禁止MTP 91](#_Toc49522498)

[15.7.3附：禁止MTP所禁止的文件列表 92](#_Toc49522499)

[16 错误码与ini 92](#_Toc49522500)

[16.1 常见错误码 92](#_Toc49522501)

[16.1.1 注册失败错误码 92](#_Toc49522502)

[16.1.2 终端操作员登录失败错误码 94](#_Toc49522503)

[16.1.3 驱动停止工作错误码 94](#_Toc49522504)

[16.1.4 导入离线策略错误码 95](#_Toc49522505)

[16.1.5 申请解密失败错误码 95](#_Toc49522506)

[16.1.6 压缩包穿透加解密错误码 96](#_Toc49522507)

[16.1.7 独立审批：终端与独立审批通信返回的值（一般会在LdApprovalEx.log体现） 97](#_Toc49522508)

[16.1.8 移动终端错误码（包括登录和处理集成审批请求的） 97](#_Toc49522509)

[16.1.9 批量加密错误码 99](#_Toc49522510)

[16.1.10 批量解密错误码 99](#_Toc49522511)

[16.1.11 应用接入系统错误码 100](#_Toc49522512)

[16.2 常用ini 101](#_Toc49522513)

[16.2.1 LdDataServer.ini 101](#_Toc49522514)

[16.2.2 LdConsole.ini 101](#_Toc49522515)

[16.2.3 User.ini 102](#_Toc49522516)

[16.2.4 ArServer.ini 102](#_Toc49522517)

[16.2.5 ArVsCfg.ini 103](#_Toc49522518)

[16.2.6 ArComm.ini 103](#_Toc49522519)

[16.2.7 Fotbit.ini 104](#_Toc49522520)

[16.2.8 ArsDbClient.ini 104](#_Toc49522521)

[16.2.9 WinTrSys.ini 105](#_Toc49522522)

[16.2.10 ArReRights.ini（详见《关于绿盾特殊功能开启授权说明（仅内部使用）.pdf》） 106](#_Toc49522523)

[16.2.11 Debug.ini（谨慎使用） 106](#_Toc49522524)

## **终端日志介绍**

|  |  |
| --- | --- |
| **日志名称** | **用途** |
| **ArWorkstation.log** | 终端ArWs.dll产生的日志，一般用于查看受控程序配置、终端版本、终端是否启动、程序是否被绿盾限制不能运行、离线策略导入问题等。 |
| **ArDriver.log** | 终端文件过滤驱动产生的日志（LdEis.sys)，一般用于查看受控程序加密问题。 |
| **ArwCapture.log** | 网络数据相关模块日志，一般用于查看端口限制，网页限制等问题。 |
| **LdSysCtrl.log** | HOOK相关模块日志，一般用于查看打印监控、打印限制、qq白名单等HOOK相关的问题 |
| **LdFileGate.log** | 服务器白名单、邮件白名单相关问题日志，一般用于查看上传和下载不加密或者不解密的问题。 |
| **LdApproval.log或LdApprovalEX.log** | 审批流程日志，一般用于查看申请解密失败或者审批不成功、文件不能解密等问题。 |
| **Ldx.log** | 外发文件打开日志，一般用于查看外发文件打不开问题。 |
| **LdImportant.log** | 错误日志，一般用于查看受控程序没有添加而导致的加密文件打不开。比如加密文件打不开，解密之后就可以打开，那一般都是受控程序没添加的问题，还有无权打开加密文件（这个是阅读权限的问题）可以直接查看错误日志。 |

# 1 透明加解密

## 1.1 透明加解密——配置日志

ArWorkstation.log（注：配置日志大部分在ArDriver.log里也能找到，这里以ArWorkstation为说明；

ArDriver.log里的配置日志如无特殊情况不再细说，ArDriver.log以现象日志为主）

### 1.1.1 受控进程的配置

**关键字：SetSafeProcess!!**

**日志示例：**

[ArWs]SetSafeProcess <0000> < WinWord.exe> <\*.\*> <\*.\*> <.doc|.docx> <WINWORD.EXE> HookType=<0000fff8> SessionId<30848>Len<65>

//受控winword.exe允许打开\*.\*；存盘\*.\*；常用文件后缀.doc|.docx；原始文件名WINWORD.EXE；允许粘贴加密的剪贴板内容（不允许粘贴则HookType=<00000000>）

### 1.1.2 EncrytType行

**关键字：EncryptType**

**日志示例：**

EncryptType = 3,ForbitScreen=0,OfflineTime=480小时+0分钟<1728000秒>,

Open=1,Copy=1,ICON=1,DecryptT=0,CtrlCode=2052

EncryptType = 3 //表示加密类型是透明加解密【1只解密不加密；3透明加解密；5打开后加密】

ForbitScreen=0 //表示不禁止截屏【0不禁止截屏；1禁止Print Screen键；2禁止所有截屏；3打开加密文件时禁止截屏】

Open=1 //早期版本，略过

Copy=0 //早期版本，略过

ICON=1 //显示加密图标

注：这里的ICON是终端图标是否显示、加密图标是否显示、不显示explorer右键菜单这3项的结合，1转换为二进制是0001

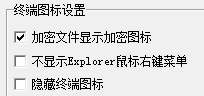
从右到左代表：

1<<0 显示加密图标

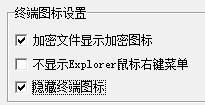
1<<1 隐藏终端图标

1<<2 隐藏explorer右键菜单

右数第一位是1，也就是说“显示加密图标”是打钩的，其余两项不打钩，即下图：



再比如ICON=3,转换二进制是011,说明是隐藏终端图标、加密文件显示加密图标，即下图：



DecryptT=1 //具有批量解密权限

CtrlCode=2052 //转换为二进制，和安全选项的其他设置（除“具有批量解密权限”外）以及高级设置一一对应

2052的二进制是1000 0000 0100，说明第2位和第11位是打钩的，也就是打钩了“允许制作打印外发文件”和“允许制作直接外发文件”

#### 1.1.2.1 注：EncryptType行CtrlCode值

表示的是操作员类型/安全选项/其他设置，以及操作员类型/高级设置的各个选项【0表示不打钩，1表示打钩】，二进制从右到左每一位对应的选项列表：

= 1<<0 //禁止ole拖拽

= 1<<1 //禁止删除文件

= 1<<2 //允许制作打印外发文件

= 1<<3 //自动同意解密

= 1<<4 //自动同意外发

= 1<<5 //表示打开网上邻居的文件不去自动加密

= 1<<6 //禁止虚拟打印到文件

= 1<<7 //禁止应用程序改名

= 1<<8 //允许explorer压缩文件

= 1<<9 //进程隐藏

= 1<<10 //关闭批量加密功能

= 1<<11 //允许制作直接外发文件

= 1<<12 //关闭批量加密记录

= 1<<13 //关闭批量解密记录

= 1<<14 //启动增强的本地备份功能

= 1<<15 //启动智能识别加密算法

= 1<<16 //启用高级加密算法

= 1<<17 //网络文件采用二代加密算法

= 1<<18 //启用缓存数据，提高速度

= 1<<19 //禁止资源管理器预览二代加密文件

= 1<<20 //在解密模式下，拷贝密文加密

= 1<<21 //支持微点杀毒软件

= 1<<22 //网络文件支持多次加密

= 1<<23 //保存网络文件不是用分页IO

= 1<<24 //二代加密的文件不清除缓存

= 1<<25 //允许批量修改文件密级

= 1<<26 //允许远程桌面拷贝本地文件

= 1<<27 //仅配置二代加密的程序才进行二代加密

= 1<<28 //禁用Linux/Mac非受控进程管道传输

= 1<<29 //不处理驱动上报的共享上的文件加解密请求

### 1.1.3 密级

**ArWorstation关键字：Level或UserName**

在UserName会显示该操作员的密级，看Level的值即可

**日志示例：**

[ArWs]TermUser TermId<1> TypeId<1> UserName<User11539> UsbOnline=<0> ComputerId=<0> GroupId=<1> Level=<0>

//终端登录的操作员名称是User11539，用户密级是0，即公开文件（0公开文件；1内部资料文件；2秘密文件；3机密文件；4绝密文件）

**ArDriver.log关键字：SecretLevel**

**日志示例：**

FileEncryptService\_02: UserId=36, GroupId=1, UserCode=1006, SecretLevel=0, EncryptType=3, EncryptWhenCopy=1, EncryptWhenOpen=1, CtrlCode=02000000, typeid=1, EncryptSection=0

UserID：操作员编号

GroupID：部门编号

UserCode：客户编号

SecretLevel：操作员密级

EncryptType：1只解密不加密，3透明加解密，5打开后加密

EncryptWhenCopy：拷贝时加密，0关1开 （Explorer复制加密）

EncryptWhenOpen：打开后加密，0关1开 （常用后缀打开后加密）

CtrlCode：对应ArWorkstation.log的EncryptType行的CtrlCode

### 1.1.4 特殊目录

**关键字：SpecialFile**

**日志示例：**

SpecialFile Send <0> <\appdata\locallow\sogoupy> //特殊目录路径 \appdata\locallow\sogoupy

SpecialFile Send <0> <\appdata\locallow\sogoupy.users> //特殊目录路径 \appdata\locallow\sogoupy.users

### 1.1.5 特殊文件后缀

**关键字：SendSpFileExtToDriver**

[ArWs]SendSpFileExtToDriver <.sldprt> CtrlCode=<0> //.sldprt后缀即不开启备份也不开启拷贝时加密

[ArWs]SendSpFileExtToDriver <.slddrw> CtrlCode=<1> //.slddrw后缀开启备份但不开启拷贝时加密

[ArWs]SendSpFileExtToDriver <.wps> CtrlCode=<8> //.wps后缀开启拷贝时加密但不开启备份

[ArWs]SendSpFileExtToDriver <.xls> CtrlCode=<9> //.xls后缀既开启备份也开启拷贝时加密

注：特殊文件后缀的CtrlCode须转换为八进制，个位代表“新建或编辑后备份”，十位代表“拷贝时加密”，所以：

0：代表不备份也不拷贝加密。

1：代表开启备份但不拷贝加密。

8：代表开启拷贝加密但不备份。

9：代表既备份也拷贝加密。

### 1.1.6 受控程序高级设置

**关键字：SendExceptFileToDriver**

[ArWs]SendExceptFileToDriver <et.exe> <\appdata\roaming\kingsoft\wps\> <\*.\*> <0>

//et.exe进程对\appdata\roaming\kingsoft\wps\目录所有后缀自动解密

附：受控程序高级设置参数列表：

<0> 自动解密

<2> 二代加密算法

<3> 备份过滤

<4> 关闭内存映射文件

<8> 不加密可执行文件

<16> 禁止阅读密文

<32> 只允许保存指定后缀的文件

<64> 禁止指定文件运行加载

<128> 排除增强网络访问

<256> 切换模式下禁止访问密文

<512> 增强型文件大小获取

<1024> 落地加密

## 1.2 透明加解密——现象日志

ArDriver.log、ArWorkstation.log、LdImportan.log

**【受控程序、非受控程序打开加密文件】**

### 1.2.1 非受控程序打开加密文件

**ArDriver.log关键字：NotConnernedProcess**

也可以直接查文件名（但路径过长可能会被截断导致查不到）

**日志示例：**

[et.exe][FFFFF8A009C69640]FsdHandleNotConcernedProcess FlushCache<\Device\HarddiskVolume1\Users\ten\Desktop\example.xlsx>

//非受控进程et.exe打开加密的example.xlsx文件

另外，如果开启错误日志，LdImportant.log会显示

进程路径：C:\Users\ten\AppData\Local\Kingsoft\WPS Office\11.1.0.9912\office6\et.exe 日志内容：[et.exe][1]不能打开文件<\Device\HarddiskVolume1\Users\ten\Desktop\ example.xlsx >

//进程et.exe不能打开加密文件

### 1.2.2 受控程序打开加密文件

**ArDriver.log关键字：Create.Fcb或Create.Fcb Exist** 通常也可以直接查文件名（但路径过长可能会被截断导致查不到）

**日志示例：**

[EXCEL.EXE][FFFFF8A009C69640][FFFFFA8001C814A0]-1-Create.Fcb Exist,EncType=3,ReadDec=1,EncWrite=1,NeedEncCopy=0,EncCopy=0,SpecPath=0,DA=00120089,CD=00000001,OP=01400060

[EXCEL.EXE][FFFFF8A009C69640][FFFFFA80042159B0]-2<00002581>File=\Device\HarddiskVolume1\Users\ten\Desktop\example.xlsx

//EXCEL.EXE打开加密的example.xlsx文件

一条打开日志**分两行显示**（可以一定程度上解决路径过长导致查询文件名经常查不到的问题），分别有-1,-2标识。

-1日志开头是进程名EXCEL.EXE，接着两个是两个句柄值不需要管，日志以Create Fcb或Create Fcb Exist为关键字。

EncType：0明文，1一代加密，2二代加密，3三代加密

ReadDec：1 读取本文件时按策略应当解密读取（只是策略上，实际还要看其他一些参数）

EncWrite：1 按策略写文件时应该加密（同样只是按策略上，并不一定就加密了）

NeedEncCopy：0拷贝时是否加密，1加密，0不加密，这个只对Explorer进程有意义

EncCopy：0 如果是1，表示当前正处于拷贝加密文件期间（一般是非受控进程拷贝加密文件或者接收加密文件到本地），不对这个文件进行加解密(此时无视ReadDec及EncWrite)

SpecPath：0本文件是否在特殊目录中，0否，1不加密目录中，2不解密目录中

DA：请求的权限，研发使用，不细说

CD：1打开，2创建，3有则打开无则创建，4覆盖原文件，5有则覆盖无则打开

OP：详细参数，研发使用，不细说

-2后面的<>是文件大小，再往后，是文件路径。文件路径并不是以C:\之类的开头，而是底层驱动路径开头。比如

C:\Users\ten\Desktop\example.xlsx 在驱动日志里显示的是\Device\HarddiskVolume1\Users\ten\Desktop\example.xlsx

### 1.2.3 写加密头

**关键字：WriteEncryptedHeader**

(2020-08-10 18:10:41)[2988]0000018296ProcessName:notepad.exe,WriteEncryptedHeader success,FileName:\Users\zhengyu\Desktop\456.txt

(2020-08-10 18:10:41)[2988]0000018296[notepad.exe][FFFFF8A00F876140][FFFFFA8007A50010]-1-Create Fcb,EncType=3,ReadDec=1,EncWrite=1,NeedEncCopy=0,EncCopy=0,SpecPath=0,DA=0012019f,CD=00000003,OP=03000060

(2020-08-10 18:10:41)[2988]0000018296[notepad.exe][FFFFF8A00F876140][FFFFFA800E12D070]-2<00000000> File=\Device\HarddiskVolume1\Users\zhengyu\Desktop\456.txt

//进程notepad.exe对\Users\zhengyu\Desktop\456.txt文件写加密头，也就是驱动层加密该文件。注：本行日志里的路径省略了分区名，如要看完整路径，可以在该日志之后两行看，因为写加密头后必定同时有打开该加密文件的日志，这个日志参考上面的【受控程序、非受控程序打开加密文件】

### 1.2.4 常用文件后缀加密

ArDriver.log ArWorkstation.log

**ArDriver.log关键字：ReportList\_AddReport:flag=2**

**ArWorkstation.log 关键字：DriverReport byFlag=<2>**

Flag：2进程关闭后加密，3马上解密，4马上删除，5马上加密，6提醒一代文件升级二代，8是需要智能加密分析进行加解密

（1）ArDriver.log驱动日志上报了一条需要在进程退出后加密这个文件的任务

[photolaunch.exe](0000000000000000)ReportList\_AddReport:flag=2, network=0, iPos=44, pid=14096, file=\Device\HarddiskVolume1\Users\zhengyu\Desktop\QQ截图20200727163904.png

（2）ArWorkstation.log终端服务进程的日志打出来有收到这条任务

[ArWs]DriverReport byFlag=<2> PID=<14096> PName<photolaunch.exe> RecordNo=<44> SrcName=<\Device\HarddiskVolume1\Users\zhengyu\Desktop\QQ截图20200727163904.png> DstName=<>

（3）加密过程



(2020-08-10 14:53:21)[2988][ArWs]DriverReport DoProcess begin encrypt!<C:\Users\zhengyu\Desktop\QQ截图20200727163904.png>

(2020-08-10 14:53:21)[2988]DoEncryptFile2 = 1,.PNG

(2020-08-10 14:53:21)[2988][ArWs]Process photolaunch.exe Close <21420>

(2020-08-10 14:53:21)[2988][ArWs]FileCache BackupFile CurrPath<C:\$Eis$Bak\>

(2020-08-10 14:53:21)[2988][ArWs]FileCache BackupFile Path CurrPath<C:\$Eis$Bak\>FileName<C:\Users\zhengyu\Desktop\QQ截图20200727163904.png.ldtmp>NewFileName<7h5LrzqSL%qLTtCSktsBsr4nczqHsvR9zQQbXsKqXI89NK19qh8xNj6hVj1Oi%AwcGBocYbPL1>

(2020-08-10 14:53:21)[2988][ArWs]AutoBackup! <C:\Users\zhengyu\Desktop\QQ截图20200727163904.png.ldtmp> <2020-07-27 16:39:07> <204>

(2020-08-10 14:53:21)[2988][ArWs]FileCache BackupFile(MoveFile) Succeed.<C:\Users\zhengyu\Desktop\QQ截图20200727163904.png.ldtmp>

(2020-08-10 14:53:21)[2988][ArWs]DoEncryptFile Local <C:\Users\zhengyu\Desktop\QQ截图20200727163904.png>Succeed.Flag=<2><14096> <250>

### 1.2.5 密级不够无权打开加密文件

ArDriver.log、LdImportant.log

**关键字：Create,Not Dec或FileLevel**

如果开启了密级判断，终端就打不开密级比自己高的文件，ArDriver.log日志将会显示该文件的密级，并提示不能打开文件，以及Create,Not Dec

如果有开启错误日志，则错误日志也会显示Create,Not Dec

**ArDriver.log日志示例：**

(2020-08-18 18:48:41)[1988]0000008828Create,Not Dec,ProcessName:et.exe,FileLevel:1,FileName:\Users\ten\Desktop\test33.xlsx

//et.exe无法打开加密文件，该文件的密级是1，也就是内部资料文件（0公开文件；1内部资料文件；2秘密文件；3机密文件；4绝密文件）

**LdImportant日志示例：**

(2020-08-18 18:48:38)[1988]进程ID:8828 进程路径:C:\Users\ten\AppData\Local\Kingsoft\WPS Office\11.1.0.9912\office6\et.exe

日志内容：[et.exe][1]不能打开文件<\Device\HarddiskVolume1\Users\ten\Desktop\test33.xlsx>

(2020-08-18 18:48:38)[1988]进程ID:8828 进程路径:C:\Users\ten\AppData\Local\Kingsoft\WPS Office\11.1.0.9912\office6\et.exe

日志内容：Create,Not Dec,ProcessName:et.exe,FileLevel:1,FileName:\Users\ten\Desktop\test33.xlsx

//et.exe无法打开加密文件，该文件的密级是1，也就是内部资料文件（0公开文件；1内部资料文件；2秘密文件；3机密文件；4绝密文件）

## 1.3 落地加密

### 1.3.1 落地加密——配置日志

ArWorkstation.log

**关键字 SendExceptFileToDriver或<1024>**

SendExceptFileToDriver <\*.\*> <\test\> <.xls|.xlsx> <1024>

SendExceptFileToDriver //高级受控关键字

<\*.\*> <\test\> <.xls|.xlsx> <1024> //配置所有进程对\test\目录.xls|.xlsx后缀落地加密

ArDriver.log

**关键字 Code=1024**

FileEncryptService\_23: FrameTotal=39,curno=34,Code=1024,process=\*.\*, Dir=\test\, FileExt=|.xls|.xlsx

FileEncryptService\_23 //受控程序高级设置

Code=1024 //表示这条配置是落地加密

process=\*.\*, Dir=\test\, FileExt=|.xls|.xlsx //落地加密的具体配置

参数说明：

<1024> 【落地加密类型】

附：受控程序高级设置参数列表：

<0> 自动解密

<2> 二代加密算法

<3> 备份过滤

<4> 关闭内存映射文件

<8> 不加密可执行文件

<16> 禁止阅读密文

<32> 只允许保存指定后缀的文件

<64> 禁止指定文件运行加载

<128> 排除增强网络访问

<256> 切换模式下禁止访问密文

<512> 增强型文件大小获取

<1024> 落地加密

### 1.3.2 落地加密——现象日志

ArDriver.log

**关键字 ReportList\_AddReport:flag=5**

[WINWORD.EXE](FFFFC6082768F170)ReportList\_AddReport:flag=5, network=0, iPos=1, pid=7664, file=\Device\HarddiskVolume1\Users\ten\Desktop\TEST\example.xlsx

//驱动上报应用层进行加密，flag=5是指马上进行加密。

ArWorkstation.log

关键字 DriverReport byFlag=<5>或DoProcess begin encrypt! Flag=<5>

[ArWs]DriverReport byFlag=<5> PID=<4492> PName<explorer.exe> RecordNo=<4> SrcName=

<\Device\HarddiskVolume1\Users\ten\Desktop\TEST\example.xlsx> DstName=<>

//收到驱动上报应用层的立即加密请求

[ArWs]DriverReport DoProcess begin encrypt!Flag=<5> <C:\Users\ten\Desktop\TEST\example.xlsx>

[ArWs]DriverReport DoProcess DoEncryptFile Local

<C:\Users\ten\Desktop\TEST\example.xlsx>Succeed.Flag=<5><4492> <62>

//启动应用层加密，加密成功

#### 1.3.2.1附：驱动上报应用层加解密的编号列表

atEncAfterProcessClose = 2,//进程关闭后加密

atDecNow = 3,//马上解密

atDelNow = 4,//马上删除

atEncNow = 5,//马上加密

atUpdateEncVersionToV7 = 6, //转换成二代加密

atSmartEnc = 8, //需要智能加密分析进行加解密

### 1.3.3 落地加密——查看例外

ArDriver.log

**关键字：FileEncryptService\_05**

[11220]0000011220FileEncryptService\_05: PIDCount=11

//说明添加例外的进程数有11个

[11220]0000011220 PID=4992

[11220]0000011220 PID=5020

[11220]0000011220 PID=5028

[11220]0000011220 PID=11220

[11220]0000011220 PID=10156

[11220]0000011220 PID=7580

[11220]0000011220 PID=13808

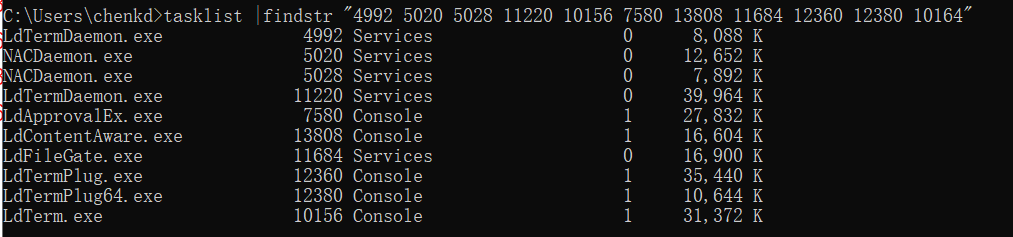
[11220]0000011220 PID=11684

[11220]0000011220 PID=12360

[11220]0000011220 PID=12380

[11220]0000011220 PID=10164

//分别是十一个例外进程的PID，可以通过任务管理器或者CMD进行查看



## 1.4 半透明加密

### 1.4.1 半透明加密——配置日志

ArWorstation.log

**关键字：SmartEnc**

SetSmartEncFile ProcessCount<1> //半透明加密进程数量

SendSmartEncAppToDriver <excel.exe> <11> //进程excel.exe配置了半透明加密，参数说明：

11【不允许模式切换】

12【允许模式切换】

SmartAppListKeep.Add <excel.exe>EncLevel<0> //参数说明：

0文件加密密级为【公开密级】

255文件加密密级为【当前操作员密级】

SendSmartEncAppToDriverMiniFilter szProcessName<excel.exe>,dwEncMode<00000003>,dwTransferType<0000000b>,bySpecifyEncLevel<0> //参数说明：

dwEncMode是3项设置的集合，其中

加密文件编辑时保持加密信息2 【不打钩为0，打钩为2】

新文件自动加密1【不打钩为1，打钩为0】

允许阅读超过操作员密级的文件4【不打钩为0，打钩为1】

所以dwEncMode<00000003>，相当于加密文件编辑时保持加密信息打钩，另外2个不打钩

dwTransferType

dwTransferType<0000000b> 【不允许模式切换】

dwTransferType<0000000c> 【允许模式切换】

bySpecifyEncLevel

bySpecifyEncLevel<0> 文件加密密级为【公开密级】

bySpecifyEncLevel<255> 文件加密密级为【当前操作员密级】

SendSmartEncToDriver <excel.exe> <\appdata\roaming\microsoft\excel\> <.xlb> <15> //半透明进程excel.exe的高级设置，参数说明：

13 关键后缀：允许切换模式时才需要设置的。也就是进程在加载这个后缀的文件时，当这个是加密的，则进程转入加密模式。

14 例外后缀：是对关键后缀的例外，一般来说是为了排除掉一些目录下的后缀与关键后缀相同的文件，加载这些文件也不按它们是否加密文件而改变加密模式。

15 例外后缀（读明文）：解密读取这些文件，以保证软件能正常运行。

### 1.4.2 半透明加密——现象日志

ArDriver.log

**关键字：SmartEnc**

[EXCEL.EXE][000009d4]SmartEncrypt\_PIDList\_Add EncMode=00000003,TransferMode=11,SpecifyEncLevel=0

//excel.exe的半透明加密配置（参考ArWorkstation.log）

[EXCEL.EXE](FFFFF8A001E88A30)FileLogList\_AddRecord ReadFile(ProcessId=2516,file=\Device\HarddiskVolume1\Users\ten\Desktop\test11.xlsx)

[EXCEL.EXE]SmartEncrypt\_PIDList\_Notice<PID=2516><bFind=1>(\Device\HarddiskVolume1\Users\ten\Desktop\TEST11.XLSX).

//半透明进程excel.exe读取\Device\HarddiskVolume1\Users\ten\Desktop\TEST11.XLSX

SmartEncrypt Update ProcessStatus to<00000002> //转入加密模式

SmartEncrypt Update ProcessStatus to<00000001> //转入不加密模式

因半透明加密组件LdSmartEnc32/64.dll不能加载等原因而导致打不开加密文件，错误日志显示：

[EXCEL.EXE][1]不能打开文件<\Device\HarddiskVolume1\Users\ten\Desktop\test11.xlsx

## 1.5 智能加密

### 1.5.1 智能加密——配置日志

#### 1.5.1.1 ArWorstation.log

**关键字SendLoginInfoToLdContentAware**

TermTypeOption ID=<6001> Setup=<1> //非敏感文件自动解密,Setup=0表示不勾选，Setup=1表示勾选

SendContentAwareFilterRuleToDriver <\users\tipray\desktop\testdir> <0> //例外目录

CContentAware::GetFilterRules Count=0 //表示没有例外目录

SendContentAwareRuleToDriver <notepad.exe> <\*.\*> <.txt> <0> //受控程序设置，表示受控程序是notepad.exe，目录\*.\*,文件后缀.txt。意思是为notepad进程在所有目录下的txt后缀文件都需要进行扫描判断是否含有敏感内容。

#### 1.5.1.2 ArDriver.log

**关键字FileEncryptService\_40**

FileEncryptService\_40: FrameTotal=1,curno=1, Code=0, process=notepad.exe, Dir=\*.\*, FileExt=.txt//当前策略总条数为1条，目前为第一条，类型为0，进程为notepad.exe，目录为\*.\*，后缀名为.txt。意思是为notepad进程在所有目录下的txt后缀文件都需要进行扫描是否含有敏感信息

#### 1.5.1.3 ContentAware.log（具体敏感策略）

**关键字：classification//策略分类数据**

智能加密策略信息-修改策略

checkRule //策略基本信息

checkExpr //定义了检测规则的逻辑运算表达式，其支持使用&, |, ()这三种运算符，定义较复杂的逻辑运算，其运算的主要元素是检测规则的id，比如正则表达式id与关键字规则id

exceptRule //例外规则，采用并集

ruleType //规则类型，1 --- 代表文件属性规则/2 --- 代表关键字规则/3 --- 代表正则表达式规则/4 --- 代表数据库指纹规则/5 ---- 代表向量机规则

exceptWord //例外条件-例外关键词

checkTimes //最少需要满足的检测次数

checkPosition //检测位置：主要针对邮件，默认需要设置"1,2,3,4"。1 --- 标题/2 --- 正文/3 --- 附件文件名 /4 --- 附件文件内容

caseSensitive //是否忽略大小写,false---不区分

Keywords //关键词/关键词组

Dbkeywords //关键词对，例如：“绝-密-2”，2代表最大间距

Regular //正则表达式

##### 1.5.1.3.1 附-常规正则表达式

|  |  |
| --- | --- |
| **常用字段** | **正则表达式** |
| IP地址 | ((?:(?:25[0-5]|2[0-4]\d|((1\d{2})|([1-9]?\d)))\.){3}(?:25[0-5]|2[0-4]\d|((1\d{2})|([1-9]?\d)))) |
| 车辆VIN码 | [a-zA-Z0-9]{17} |
| 车牌号 | [京津沪渝冀豫云辽黑湘皖鲁新苏浙赣鄂桂甘晋蒙陕吉闽贵粤青藏川宁琼使领A-Z]{1}[A-Z0-9]{5}[A-Z0-9挂学警港澳]{1} |
| 大陆护照号 | [GgEe]{1}\d{8} |
| 港澳通行证号 | [HMhmCcWw]{1}([0-9]{10}|[0-9]{8}) |
| 家庭地址 | (北京市|天津市|上海市|重庆市|河北省|山西省|辽宁省|吉林省|黑龙江省|江苏省|浙江省|安徽省|福建省|江西省|山东省|河南省|湖北省|湖南省|广东省|海南省|四川省|贵州省|云南省|陕西省|甘肃省|青海省|台湾省|内蒙古|广西|西藏|宁夏|新疆|香港|澳门)[^\s]{1,60}(县|区|镇|乡|路|街|村|屯)[^\s]{1,10}(组|号|区)?[0-9]{0,4}室? |
| 身份证号 | [1-9]\d{5}(18|19|([2]\d))?\d{2}((0[1-9])|(10|11|12))(([0-2][1-9])|10|20|30|31)(\d{3}|\d{2})[0-9Xx] |
| 手机号码 | [1]((3[0-9])|(4[579])|(5[0-3|5-9])|(7[0|1|3|5-8])|(8[0-9]))[0-9]{8} |
| 税务登记证号 | \d{15} |
| 统一社会信用代码 | [1-9A-GY]{1}[1239]{1}[1-5]{1}[0-9]{5}[0-9A-Z]{10} |
| 银行卡号 | ([0-9]{16}|[0-9]{19}) |
| 邮件地址 | [\w!#$%&''\*+/=?^\_`{|}~-]+(?:\.[\w!#$%&''\*+/=?^\_`{|}~-]+)\*@(?:[\w](?:[\w-]\*[\w])?\.)+[\w](?:[\w-]\*[\w])? |
| 中国姓名 | (李|王|张|刘|陈|杨|赵|黄|周|吴|徐|孙|胡|朱|高|林|何|郭|马|罗|梁|宋|郑|谢|韩|唐|冯|于|董|萧|程|曹|袁|邓|许|傅|沈|曾|彭|吕|苏|卢|蒋|蔡|贾|丁|魏|薛|叶|阎|余|潘|杜|戴|夏|钟|汪|田|任|姜|范|方|石|姚|谭|廖|邹|熊|金|陆|郝|孔|白|崔|康|毛|邱|秦|江|史|顾|侯|邵|孟|龙|万|段|雷|钱|汤|尹|黎|易|常|武|乔|贺|赖|龚|文|庞|樊|兰|殷|施|陶|洪|翟|安|颜|倪|严|牛|温|芦|季|俞|章|鲁|葛|伍|韦|申|尤|毕|聂|丛|焦|向|柳|邢|骆|岳|齐|尚|梅|莫|庄|辛|管|祝|左|涂|谷|祁|时|舒|耿|牟|卜|路|詹|关|苗|凌|费|纪|靳|盛|童|欧|甄|项|曲|成|游|阳|裴|席|卫|查|屈|鲍|位|覃|霍|翁|隋|植|甘|景|蒲|单|包|司|柏|宁|柯|阮|桂|闵|欧|阳|解|强|柴|华|车|冉|房|边|辜|吉|饶|刁|瞿|戚|丘|古|米|池|滕|晋|苑|邬|臧|畅|宫|来|缪|苟|全|褚|廉|简|娄|盖|符|奚|木|穆|党|燕|郎|邸|冀|谈|姬|屠|连|郜|晏|栾|郁|商|蒙|计|喻|揭|窦|迟|宇|敖|糜|鄢|冷|卓|花|仇|艾|蓝|都|巩|稽|井|练|仲|乐|虞|卞|封|竺|冼|原|官|衣|楚|佟|栗|匡|宗|应|台|巫|鞠|僧|桑|荆|谌|银|扬|明|沙|薄|伏|岑|习|胥|保|和|蔺)[\u4e00-\u9fa5]{1,3} |
| 织机构代码 | [a-zA-Z0-9]{8}-[a-zA-Z0-9] |
| 座机号码 | (0\d{2}-?\d{8})|(\(0\d{2}\)\d{8})|(0\d{3}-?\d{7,8})|(\(0\d{3}\)\d{7,8}) |

### 1.5.2 敏感策略案例分析

#### 1.5.2.1 关键字规则

{

"id":3, //主键

"ruleType":"2", //规则类型，“2”表示关键字规则

"exceptWord":"qq,cz1-cz2,kw", //例外关键字或词组

"checkTimes":3, //匹配计数

"checkPosition":"1,2,3,4", //匹配位置：1-主题；2-正文；3-附件名；4-附件正文

"caseSensitive":1, //是否区分大小写

"keywords":"ww,czk1-czk2-czk3,kww", //关键字或词组

"dbkeywords":"k1-k2-12,k11-k12-32", //关键字对

"createTime": 1527496084007 //规则编辑时间

}

#### 1.5.2.2 正则表达式规则

{

"id":4, //主键

"ruleType":"3", //规则类型，“3”表示正则规则

"exceptWord":"qq,cz1,cz2,kw", //例外关键字或词组

"checkTimes":3, //匹配计数

"checkPosition":"1,2,3,4", //匹配位置：1-主题；2-正文；3-附件名；4-附件正文

"regular":"正则表达式", //正则表达式

"createTime": 1527496084007 //规则编辑时间

}

### 1.5.3 智能加密——现象日志

#### 1.5.3.1 ArDriver.log

**关键字：ReportList\_AddReport**

ReportList\_AddReport:flag=8,network=0,iPos=1,pid=5152,file=\Device\HarddiskVolume1\Users\tipray\Desktop\TEST (2).txt

上报需要进行敏感信息扫描的文件，file指出具体路径和文件名，flag=8（上报类型为需要分析文件内容）

#### 1.5.3.2 Arworstation.log

**关键字：DriverReport**

DriverReport byFlag=<8> PID=<5152> PName<notepad.exe> RecordNo=<1> SrcName=<\Device\HarddiskVolume1\Users\tipray\Desktop\TEST (2).txt> DstName=<>

//在进程notepad退出之后，开始对SrcName字段里的文件进行内容分析扫描

DriverReport Need Analyze File <C:\Users\tipray\Desktop\TEST (2).txt>

//需要进行扫描的文件

SendFileAnalyze0101 To LdContentAware..

//将上面TEST (2).txt发到ldcontentaware进程进行扫描

CContentAware::Recv<0102>

CContentAware::DealFrame0102..

CContentAware::AddEncDecWork<0><C:\Users\tipray\Desktop\TEST (2).txt>//

收到ldcontentaware的结果，1是解密文件，0是加密文件。

CContentAware EncryptFile Fail<5><C:\Users\tipray\Desktop\TEST (2).txt>!! //

因为本身是加密文件，所以加密失败。

#### 1.5.3.3 ContentAware.log（接受上报，进行分析文件内容）

**关键字：CWorkManager和DoAnalyzeWork**

CWorkManager AddNewWork<0>insert into FileAnalyzeTask(Type,FilePath,FileSize,InsertTime,Flag,CheckSum) values(1,'C:\Users\tipray\Desktop\TEST (2).txt','0','07e4-07-21 14:53:54',0,'d0c6c125993552759df6ad865518ebcd')

//获取到新任务

CWorkManager GetFileSize Sus.<00000000,00000008><C:\Users\tipray\Desktop\TEST (2).txt>

//获取文件大小和文件路径

DoAnalyzeWork,BeginAnalyze<C:\Users\tipray\Desktop\TEST (2).txt>

//开始分析文件

DoAnalyzeWork,Analyze End<C:\Users\tipray\Desktop\TEST (2).txt>

//结束分析

RemoveSameCheckSumWorks<0><delete from FileAnalyzeTask where CheckSum='d0c6c125993552759df6ad865518ebcd'>

//结束任务

DoAnalyzeWork,Result<0><C:\Users\tipray\Desktop\TEST (2).txt>

//任务结果。Result=0。1是包含敏感信息，0是不包含，ldcontentaware将这个结果发到终端服务进程，由终端服务进程进行加解密

#### 1.5.3.4 ArWorkstation.log(服务端接收ldcontentaware结果后的加解密结果)

**关键字：AddEncDecWork**

DriverReport Need Analyze File <C:\Users\tipray\Desktop\TEST (2).txt>

//收到驱动层上报的需要分析处理的文件

CContentAware::AddEncDecWork <0><C:\Users\tipray\Desktop\TEST (2).txt>

//增加加解密任务，AddEncDecWork的值0加密，1解密

CContentAware EncryptFile Sus<C:\Users\tipray\Desktop\TEST (2).txt>

//加密成功，EncryptFile Sus代表成功，Fail<5>代表文件本身已是加密文件，所以加密失败

CContentAware DecryptFile Sus<C:\Users\tipray\Desktop\TEST (2).txt>

//解密成功，DecryptFile Sus

## 1.6 应用层加解密

### 1.6.1全盘加密

ArWorkstation.log

**关键字：**

（1）DoAction 或 ScanToEncrypt：开始执行全盘加密动作

（2）DoEncryptFile2 ：全盘加密后缀类型

（3）$Eis$Bak：批量加密和全盘加密的缓存目录

**相关日志分析：**

[ArWs]DoAction ScanToEncrypt==>m\_bEncrypt[TRUE] //执行全盘加密动作，TRUE代表执行加密

开始全盘扫描...

[2196]DoEncryptFile2 = 1,.DOC // 开始扫描，扫描后缀 .doc，对该后缀执行加密

[2196][ArWs]FileCache BackupFile CurrPath<C:\$Eis$Bak\> //全盘加密缓存目录

[2196][ArWs]FileCache BackupFile Path CurrPath<C:\$Eis$Bak\>FileName<C:\Users\A\Desktop\1.doc.ldtmp>NewFileName<7h5LrzqSL%qLXshauS4nczqHsvR9zInwcvRKit4lsvp9> //1.doc符合全盘加密要求，先备份缓存到$Eis$Bak，然后再进行加密

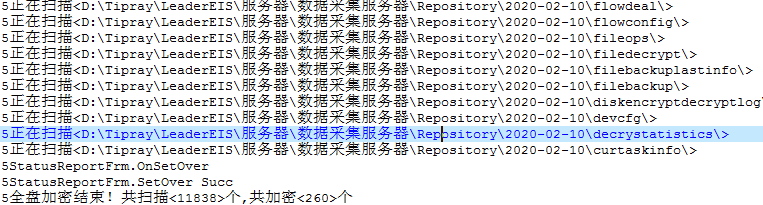
[2196]DoEncryptFileExW<C:\Users\A\Desktop\1.doc><0>

[2196]全盘扫描加密<C:\Users\A\Desktop\1.doc> //缓存已备份，开始执行对1.doc文件进行加密

全盘扫描结束, 耗时11秒, 文件总个数:11838, 待加密文件个数:265, 加密成功文件数:260

//扫描结束后，有相关的扫描报表

PS：如果想要查看具体的扫描过程以及扫描加密内容，可查看LdTerm.log



**案例分析：**

某客户对终端设置了全盘加密，然后设置的加密后缀为.doc|.docx|.xls等等，然后全盘加密之后，客户反馈好一些.doc文件没有被加密上，怀疑绿盾全盘加密不全？

**排查分析：**

1、先了解下客户的环境、以及全盘加密的时间、未能加密的文件名称等等信息；

2、获取日志，查看ArWorkstation.log，可以先看看全盘加密的结果，搜索关键字“全盘”，查看下是否有全盘扫描结束的字样以及报表信息；

3、如果相应时间点没有看到全盘扫描结束的字样，那说明全盘加密还没结束，所以可能还没加密上，也可以打开LdTerm.log搜索下那个未被加密的文件名，看看是否已经扫描到这个文件了；

4、假设已经有看到全盘扫描结束的字样，但是文件还没加密上，可以在ArWorkstation.log搜索文件名，看看是否有对这个文件进行加密，如果有，看看是否有加密失败的代码；如果没有那就检查下LdTerm.log；

5、常情况下，第四点确认后，就能知道为什么没加密上，如果还没，再具体获取日志分析下。

### 1.6.2 全盘解密

ArWorkstation.log

**关键字：**

（1）Action 或 ScanToEncrypt：开始执行全盘加密动作

（2）DoEncryptFile2 ：全盘加密后缀类型

（3）$Eis$Bak：批量加密和全盘加密的缓存目录

**相关日志分析：**

[ArWs]DoAction ScanToEncrypt==>m\_bEncrypt[FALSE] //执行全盘解密动作，FALSE代表执行解密

[2196]开始全盘扫描...

[2196]DoDecryptFileW<C:\autoexec.bat><5> //解密文件失败，5代表文件已经是明文

[2196]DoDecryptFileW<C:\LdDownloadDir\zd\_gfzq\risk.txt><0> //解密文件成功

[2196]DoDecryptFileW<C:\Users\杰生\Desktop\2.xls><7> //解密文件失败，7代表文件被占用

全盘扫描结束, 耗时2秒, 文件总个数:14779, 待解密文件个数:9, 解密成功文件数:3

//扫描结束后，有相关的扫描报表

PS：如果想要查看具体的扫描过程以及扫描解密内容，可查看LdTerm.log

### 1.6.3 批量加解密

关于批量加解密的成功以及失败，其实可以有两个方法可以看到，一个是通过实时日志，一个是通过ArWorkstation.log日志进行查看，但是不管是通过什么方式查看，都需要先了解下，批量加解密的结果代码，以下分别是批量加解密对应的代码情况。

#### 1.6.3.1 批量加密失败代码

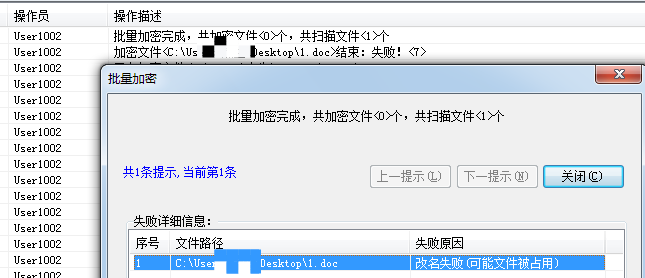
|  |  |
| --- | --- |
| 错误代码 | 原因 |
| 1 | 无法打开文件，文件不存在 |
| 2 | 读原始文件失败，//出现这个情况，也有可能是这个文件内容是空 0字节。 还有一个原因是加密类型可能是 打开文件后加密。  ArWorkstation EncryptType=5 |
| 3 | 文件夹没有权限写入 |
| 5 | 文件已经是密文，无需加密 |
| 6 | 无法加密，需要开启高级加密功能才行 （可能勾选了仅配置二代加密，倒是无法进行二代加密，可以设置LdTermdaemon.ex针对指定后缀二代加密） |
| 7、8、9 | 文件改名失败（如文件被占用），假设是三代加密，批量加密失败代码8，有可能是受控了SearchProtocol |
| 101 | 路径太长 |

#### 1.6.3.2 批量解密失败代码

|  |  |
| --- | --- |
| 错误代码 | 原因 |
| 1 | 无法打开文件，文件不存在 |
| 2 | 读原始文件失败，//出现这个情况，也有可能是这个文件内容是空 0字节。 还有一个原因是加密类型可能是 打开文件后加密。  ArWorkstation EncryptType=5 |
| 4/100 | 密钥错误 |
| 5 | 文件已经是明文，无需解密  （特殊案例：若文件确实是密文，还提示错误码5，检查下是否受控LdTermdaemon.exe进程，或者操作员类型是否“打开文件后加密”的类型） |
| 7、8、9 | 文件改名失败（如文件被占用） |
| 15 | 改名或者复制失败 |
| 17 | 拷贝失败 （可能权限问题）空间不足 |
| 18 | 读取异常了 |
| 101 | 无法打开文件，文件存在，其他原因（如文件被占用） |
| 242 | 密钥错误， 密钥序号0； 要用工具解密，联系研发 |

#### 1.6.3.3 关于批量加解密日志查看方法

（1）查看实时日志：



大家可根据实时日志提示的具体内容，然后对应上面的错误代码，来排查问题。

（2）查看ArWorkstation.log日志

A、批量解密日志，搜索关键字“DoBatchDecrypt”

DoBatchEncrypt.. //执行批量加密动作

ScanToEncrypt<C:\Users\AA\Desktop\1.doc> //扫描加密

DoEncryptFile2 = 1,.DOC //加密的后缀doc

Encrypt File <C:\Users\AA\Desktop\1.doc> fail. Result=<7> UsedTime=<874> //加密文件失败，错误代码7

B、批量加密日志，搜索关键字“DoBatchEncrypt”

DoBatchDecrypt.. //执行批量解密动作

ScanToDecrypt <C:\Users\AA\Desktop\123.txt> //对该文件进行解密

Decrypt file <C:\Users\AA\Desktop\123.txt> fail. Result=<5> AdDecrypt=<0> LastError=<0> ErrorCode=<10006> //解密该文件失败，错误代码5，是明文无需解密

### 1.6.4 驱动上报应用层加解密

正常情况，如果是驱动上报应用层进行主要会涉及到两个日志，分别是ArWorkstation.log以及ardriver.log日志，其中自动解密、常用文件后缀，特殊目录解密，落地加密等等，所涉及到的策略都基本上是一致的，所以具体要看上报的内容来分析下是哪个策略导致的驱动上报应用加解密；

首先我们先把加密的几个flag值理清楚下，只要理清楚这几个数值，就能大概清楚是哪个功能做的动作；

**日志查看关键字：**打开ardriver.log日志，搜素指定文件名并对应上被加密或者解密的时间点，然后看看下上下左右是否有以下这条记录“ReportList\_AddReport:flag=X”，注意看flag的值，

以下是flag的值以及对应的可能功能：

|  |  |  |
| --- | --- | --- |
| Flag值 | 代表意思 | 可能用到的功能 |
| 2 | 进程关闭后加密 | 常用后缀，或个别软件需进程结束后才能加密上 |
| 3 | 马上解密 | 自动解密、特殊目录 |
| 4 | 马上删除 | 暂时不需要了解 |
| 5 | 马上加密 | 落地加密 |
| 6 | 提醒一代文件升级二代 | 兼容一代升级 |
| 8 | 是需要智能加密分析进行加解密 | 智能加密 |

案例分析：

（1）案例1：假设2.doc文件，打开关闭后自动加密了，我们如果通过日志来分析它的加密过程：

第一步：打开ardriver.log，搜索“AddReport”，然后查看2.doc的日志，如下显示：

[WINWORD.EXE]ReportList\_AddReport:flag=2, network=0, iPos=5, pid=10948, file=\Device\HarddiskVolume1\Users\AA\Desktop\2.doc //winword.exe 打开了这个2.doc文件，并上报应用层去执行加密，关闭进程后加密；

第二步：打开ArWorkstation.log，搜索“DriverReport ”，然后对应上文件名和时间点，日志如下显示：

DriverReport byFlag=<2> PID=<10948> PName<WINWORD.EXE> RecordNo=<5> SrcName=<\Device\HarddiskVolume1\Users\AA\Desktop\2.doc> //驱动上报，对2.doc文件进行加密，flag=2，说明进程关闭后再加密上；

DriverReport DoProcess begin encrypt!<C:\Users\AA\Desktop\2.doc> //应用层对2.DOC文件进行加密

（2）案例2：某文件1.doc原来是加密的，然后突然刷新下就解密了，如果分析？

其实碰到文件突然被解密，一般可以怀疑下，是否是设置了自动解密或者特殊目录导致的，不过我们要先看下驱动那边是否有下发解密的动作，如果有就好确定了，具体分析如下：

第一步：打开ardriver.log，搜索“文件名”或者“AddReport”，然后根据时间点和文件名一一对应上日志，如下显示：[explorer.exe]ReportList\_AddReport:flag=3, network=0, iPos=7, pid=2188, file=\Device\HarddiskVolume1\Users\AA\Desktop\1.doc

//explorer进程，上报自动解密，flag=3，说明这个解密动作是驱动这边上报给应用层的；

第二步：打开ArWorkstation.log，搜索“文件名”或者“DriverReport ”，然后对应上文件名和时间点，日志如下显示：



第三步：可以确定该文件是绿盾这边进行的解密，这时候一般也就两个两个，可能是设置了特殊目录或者有设置自动解密，我们可以分别再进一步查下ArWorkstation.log，分别搜索以下关键字：

A、怀疑特殊目录：搜索关键字“SpecialFile ”，日志如下：

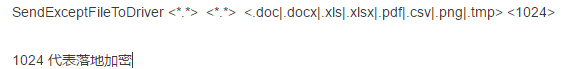
SpecialFile Send <0> <1.doc> //发现是特殊特殊目录1.doc，导致文件解密的；

B、怀疑设置了自动解密：搜索关键字“SendExceptFileToDriver”，这个是受控高级设置的日志，查到结果如下：

<winword.exe> <\*.\*> <.doc> <0> //Winword.exe 针对所有目录下的.DOC文件自动解密

小结：

通过flag的值，在配合以上的表格，基本上可以判断出哪个文件被加密或者解密，是哪个功能来实现的，比如flag=5，我们可以在ArWorkstation.log搜索关键字“SendExceptFileToDriver”，如果有某进程对某后缀设置了高级策略1024，那就是落地加密；



## 1.7 压缩包穿透加解密日志

### 1.7.1 压缩包穿透加解密——配置日志

ArWorkstation.log

**关键字SysStrOption**

**例：**

Download file succeed.<SysStrOption> UsedTime=<31> FileSize=<3720>BCProtectFiles::UpdateFile1(SysStrOption.db)!

SaveModifyTime.<SysStrOption> <2020-07-31 17:42:39> <2020-07-31 17:42:42>

说明：策略下发成功，并将策略更新到数据库表SysStrOption.db中

关键字.zip|.rar|.7z|、Setup DWORD

**例：**

SysStrOption GroupId=0, ComputerId= 10410, ID=<4040> Setup String=<.zip|.rar|.7z|> Setup DWORD=<1>

SysStrOption GroupId=0, ComputerId= 10410, ID=<4039> Setup String=<.zip|.rar|.7z|> Setup DWORD=<1>

SysStrOption GroupId=0, ComputerId= 10410, ID=<4041> Setup String=<.zip|.rar|.7z|> Setup DWORD=<1>

ID=<4040> //表示全盘加密时

ID=<4039> //表示批量加密时

ID=<4041> //表示解密文件时（包括批量解密和申请解密）

Setup String=<.zip|.rar|.7z|> //这对小“<>”里存放的数据串是勾选的压缩包后缀

Setup DWORD=<1> //表示启用

Setup DWORD=<0> //表示不启用

### 1.7.2 压缩包穿透加解密——现象日志

PenetratrEnDecryptionW.log

**关键字 Compress或文件名**

(2020-08-16 17:14:08)[INFO ][7984][PenetrateEncrypt]

Start GetUnPackTempDrive..<C:\Users\ten\Desktop\0812.zip>;

(2020-08-16 17:14:08)[INFO ][7984][PenetrateEncrypt]End GetUnPackTempDrive:<C:>;File Size:<0.02 MB>;Free Space:<3692.27 MB>

(2020-08-16 17:14:08)[INFO ][7984][PenetrateEncrypt]Do UnPackage Path<C:\171408105\>..

(2020-08-16 17:14:08)[INFO ][7984][PenetrateEncrypt]Do UnCompressFile<C:\Users\ten\Desktop\0812.zip>

(2020-08-16 17:14:08)[INFO ][7984][PenetrateEncrypt]UnCompressFile End.

(2020-08-16 17:14:08)[INFO ][7984][PenetrateEncrypt]Do RemoveRight after<C:\171408105>

(2020-08-16 17:14:08)[INFO ][7984][PenetrateEncrypt]GetScanZipPath findfirst<test11.xlsx>

(2020-08-16 17:14:08)[INFO ][7984][PenetrateEncrypt]File Real Path<C:\Users\ten\Desktop\0812.zip:test11.xlsx>

(2020-08-16 17:14:08)[INFO ][7984][PenetrateEncrypt]Do ArWs DecryptFile Suc.<C:\171408105\test11.xlsx>

(2020-08-16 17:14:08)[INFO ][7984][PenetrateEncrypt]GetScanZipPath findfirst<test22.xlsx>

(2020-08-16 17:14:08)[INFO ][7984][PenetrateEncrypt]File Real Path<C:\Users\ten\Desktop\0812.zip:test22.xlsx>

(2020-08-16 17:14:08)[INFO ][7984][PenetrateEncrypt]Do ArWs DecryptFile Suc.<C:\171408105\test22.xlsx>

(2020-08-16 17:14:08)[INFO ][7984][PenetrateEncrypt]All UnCompressFile Over, EncrypFileNum:<2>

(2020-08-16 17:14:08)[INFO ][7984][PenetrateEncrypt]Do CompressFile<C:\171408105\0812.tmp>...

(2020-08-16 17:14:09)[INFO ][7984][PenetrateEncrypt]CompressFile Suc.<C:\171408105\0812.tmp>,Ext<.zip>

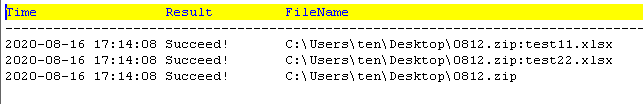
(2020-08-16 17:14:09)[INFO ][7984][PenetrateEncrypt]BackupFile<C:\Users\ten\Desktop\0812.zip>,iRe<0>

(2020-08-16 17:14:09)[INFO ][7984][PenetrateEncrypt]CompressFile clear temp<C:\171408105\>;ExitCode<0>;GetLastError<0>

此外，在ArDriver.log里可以看到，解密后，结果先放在本地\BatchLog\中：

(2020-08-16 17:14:09)[7984]0000007984[00001f30]NM Write,From 00000000,Len:00000400 \InetPub\ftproot\Tipray\LdTerm\BatchLog\20200816171407\_0001.log

之后再将结果上传到数据采集服务器上：

(2020-08-16 17:16:07)[7984]0000002616[00000a38]NM Write,From 00000000,Len:00000413 \Program Files\Tipray\LeaderEIS\服务器\数据采集服务器\DataFile\2020-08-16\BatchDecryptLog\20200816171407\_0001.log.tmp

### 1.7.3 压缩包穿透加解密——错误码

|  |  |  |
| --- | --- | --- |
| **参数名** | **错误码** | **中文说明** |
| TR\_SUCCEED | 0 | //正常，没有错误 |
| TR\_OPEN\_FILE\_FAIL | 10001 | //打开文件失败 |
| TR\_FILE\_TOO\_SMALL | 10002 | //文件是空的 |
| TR\_FILE\_TYPE\_ERROR | 10004 | //压缩失败 |
| TR\_FILE\_ALREADY\_ENCRYPT | 10005 | //文件已经是加密的 |
| TR\_FILE\_ALREADY\_DECRYPT | 10006 | //文件已经是解密的 |
| TR\_FILE\_ENCRPT\_ERROR | 10007 | //加密过程错误 |
| TR\_CREATE\_FILE\_ERROR | 10008 | //创建文件失败 |
| TR\_CREATE\_ENCRYPT\_HEAD\_FAIL | 10009 | //创建加密头失败 |
| TR\_WRITE\_FILE\_FAIL | 10010 | //写文件失败 |
| TR\_READ\_FILE\_FAIL | 10011 | //读文件失败 |
| TR\_MOVE\_FILE\_FAIL | 10012 | //改名失败 |
| TR\_DELETE\_TEMP\_FILE\_FAIL | 10013 | //删除临时文件失败 |
| TR\_GET\_KEY\_FAIL | 10014 | //密钥错误 |
| TR\_FILE\_DECRYPT\_ERROR | 10015 | //解密过程失败 |
| TR\_FILE\_VERSION\_NOT\_AVAILABLE | 10016 | //该版本尚未支持本操作 |

# 2 CoreConfig配置日志

ArWorkstation.log

**关键字：RegSetKeyValue或CoreConfig或PN\_0**

**例：**

Get Config,select TermId,GroupID,ProcessName,Config,Filter from CoreConfig where GroupID=65535

//数据库查询针对所有分组设置的CoreConfig策略（注：目前只能对整个公司生效，即所有分组）

RegSetKeyValue Sus<PN\_0><searchapp.exe> //PN\_0表示第一条策略，进程是searchapp.exe；

RegSetKeyValue Sus<TYPE\_0><2> //<2>数字2表示对该进程searchapp.exe采用过滤方式处理；

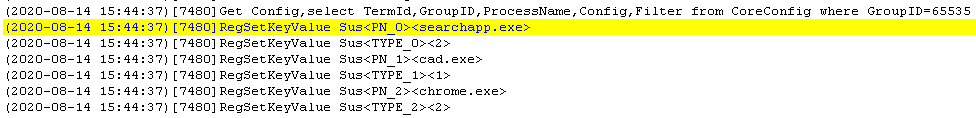
RegSetKeyValue Sus<PN\_1><cad.exe> //PN\_1表示第二条策略，且处理的进程是cad.exe；

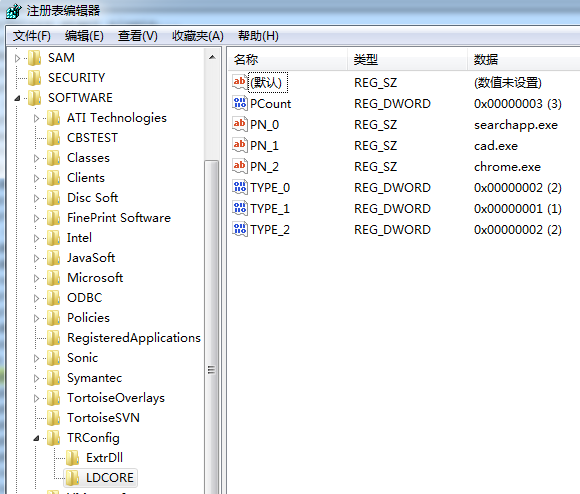
RegSetKeyValue Sus<TYPE\_1><1> //<1>数字1表示对该进程cad.exe采用新注入方式处理

**注：**以上日志是指获取到CoreConfig.db的设置，最终有没有生效要看注册表以下位置的信息：

\HKEY\_LOCAL\_MACHINE\SOFTWARE\TRConfig\LDCORE

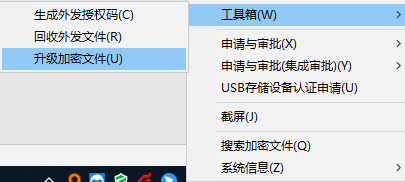
可以看到日志和注册表的信息是一一对应的：





# 3 升级加密文件

绿盾从普通版本升级到三代驱动版本，一代加密文件需要升级为二代或三代加密文件（具体是二代还是三代，是看版本以及配置，6.00以上版本默认是用三代加密算法，但可以选择关闭三代加密算法）。升级方法是在终端左键菜单的工具箱—升级加密文件，添加目录或文件，设置好备份目录，再点击开始升级即可。升级加密文件的后台过程，是先将原文件备份到指定目录（默认C:\Backup），再解密原文件，然后加密成二代或三代加密文件。





**升级加密文件的常用日志如下：**

ReEnc.log

**关键字：NM To SP、Dec File、Enc File**

## 3.1 升级成功示例

(2020-08-18 14:50:57)----------------------GlobeEnumThread Start--------------------

(2020-08-18 14:50:57)DealThreadBegin <0>

(2020-08-18 14:50:57)ENCTYPE <NM To SP>

(2020-08-18 14:50:57)AddTail<D:\test\《完美应用Ubuntu》2009年新年整合版.pdf>

(2020-08-18 14:50:57)FindAndReEncFile Enter:<D:\test\《完美应用Ubuntu》2009年新年整合版.pdf>

(2020-08-18 14:50:57)FileExt<.pdf>

(2020-08-18 14:50:57)IsNMEncryptFileByFileEx<0><0><D:\test\《完美应用Ubuntu》2009年新年整合版.pdf>

(2020-08-18 14:50:57)BackUpFile,Sus File <D:\test\《完美应用Ubuntu》2009年新年整合版.pdf><C:\Backup\200818\145057\D\test\《完美应用Ubuntu》2009年新年整合版.pdf>

(2020-08-18 14:50:58)Dec File Sus<0><D:\test\《完美应用Ubuntu》2009年新年整合版.pdf>

(2020-08-18 14:50:58)Enc File Sus<0><D:\test\《完美应用Ubuntu》2009年新年整合版.pdf>

(2020-08-18 14:50:58)DealThreadEnd <0><1><1><0><172>

(2020-08-18 14:51:15)End Process...

NM To SP //表示一代加密文件（84 7D）升级到三代驱动加密文件，可以是二代算法（87 7D）或三代算法（88 7D）的加密文件，具体看）

BackUpFile,Sus //备份原文件成功

Dec File Sus //解密成功

Enc File Sus //加密成功

## 3.2 升级失败示例

(2020-08-18 16:08:07)----------------------GlobeEnumThread Start--------------------

(2020-08-18 16:08:07)DealThreadBegin <0>

(2020-08-18 16:08:07)ENCTYPE <NM To SP>

(2020-08-18 16:08:07)AddTail<D:\test\《完美应用Ubuntu》2009年新年整合版.pdf>

(2020-08-18 16:08:07)FindAndReEncFile Enter:<D:\test\《完美应用Ubuntu》2009年新年整合版.pdf>

(2020-08-18 16:08:07)FileExt<.pdf>

(2020-08-18 16:08:07)IsNMEncryptFileByFileEx<0><0><D:\test\《完美应用Ubuntu》2009年新年整合版.pdf>

(2020-08-18 16:08:07)BackUpFile,Sus File <D:\test\《完美应用Ubuntu》2009年新年整合版.pdf><C:\Backup\200818\160807\D\test\《完美应用Ubuntu》2009年新年整合版.pdf>

(2020-08-18 16:08:07)Dec File Fail<9><D:\test\《完美应用Ubuntu》2009年新年整合版.pdf>

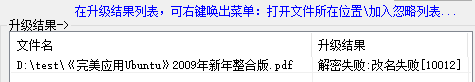
(2020-08-18 16:08:07)DealThreadEnd <0><1><1><1><0>

NM To SP //表示一代加密文件（84 7D）升级到三代驱动加密文件，可以是二代算法（87 7D）或三代算法（88 7D）的加密文件，具体看）

BackUpFile,Sus //备份原文件成功

Dec File Fail<9> //解密失败错误代码9，表示改名失败，可能是文件被占用的原因

注：如果升级失败，在升级加密文件窗口的结果列表里会显示每个文件的失败原因和错误码，ReEnc.log里也会显示错误码，但错误码不同。原因是两个地方调用的接口不一样，结果列表的错误码是调用压缩包穿透加解密接口，ReEnc.log是批量加解密接口。





## 3.3 附:升级加密文件错误码列表（同压缩包穿透加解密和批量加解密错误码）

|  |  |  |
| --- | --- | --- |
| **升级结果列表的错误码** | | |
| **参数名** | **错误码** | **中文说明** |
| TR\_SUCCEED | 0 | //正常，没有错误 |
| TR\_OPEN\_FILE\_FAIL | 10001 | //打开文件失败 |
| TR\_FILE\_TOO\_SMALL | 10002 | //文件是空的 |
| TR\_FILE\_TYPE\_ERROR | 10004 | //压缩失败 |
| TR\_FILE\_ALREADY\_ENCRYPT | 10005 | //文件已经是加密的 |
| TR\_FILE\_ALREADY\_DECRYPT | 10006 | //文件已经是解密的 |
| TR\_FILE\_ENCRPT\_ERROR | 10007 | //加密过程错误 |
| TR\_CREATE\_FILE\_ERROR | 10008 | //创建文件失败 |
| TR\_CREATE\_ENCRYPT\_HEAD\_FAIL | 10009 | //创建加密头失败 |
| TR\_WRITE\_FILE\_FAIL | 10010 | //写文件失败 |
| TR\_READ\_FILE\_FAIL | 10011 | //读文件失败 |
| TR\_MOVE\_FILE\_FAIL | 10012 | //改名失败 |
| TR\_DELETE\_TEMP\_FILE\_FAIL | 10013 | //删除临时文件失败 |
| TR\_GET\_KEY\_FAIL | 10014 | //密钥错误 |
| TR\_FILE\_DECRYPT\_ERROR | 10015 | //解密过程失败 |
| TR\_FILE\_VERSION\_NOT\_AVAILABLE | 10016 | //该版本尚未支持本操作 |

|  |  |
| --- | --- |
| **ReEnc.log的错误码** | |
| **错误码** | **原因** |
| 1 | 无法打开文件，文件不存在 |
| 2 | 读原始文件失败，//出现这个情况，也有可能是这个文件内容是空 0字节。 还有一个原因是加密类型可能是 打开文件后加密。  ArWorkstation EncryptType=5 |
| 4/100 | 密钥错误 |
| 5 | 文件已经是明文，无需解密  （特殊案例：若文件确实是密文，还提示错误码5，检查下是否受控LdTermdaemon.exe进程，或者操作员类型是否“打开文件后加密”的类型） |
| 7、8、9 | 文件改名失败（如文件被占用） |
| 15 | 改名或者复制失败 |
| 17 | 拷贝失败 （可能权限问题）空间不足 |
| 18 | 读取异常了 |
| 101 | 无法打开文件，文件存在，其他原因（如文件被占用） |
| 242 | 密钥错误， 密钥序号0； 要用工具解密，联系研发 |

# 4 Linux日志

## 4.1 tr\_service.log

tr\_service.log是记录linux终端服务运行状态的日志，日志文件路径是/var/log/trld.log

常用日志信息：

### 4.1.1 Linux加密客户端版本

**关键字：**Terminal Version 客户端版本

**关键字：**Server Version 服务器版本

### 4.1.2 用户登录

**关键字：**

Now ExecAutoLogin [14625]!!!!!! //自动登录终端编号14625

HandleResponseCheckLogin::A user has logged in [7][14625]!!!!! //用户登录信息，14625已登录

### 4.1.3 客户端与服务器链接状态

**（1）终端在线状态**

**关键字：keepOnlineAlive:reccv=[OK!][[7]**

**（2）终端离线状态：**

日志片段：

connecting server [192.168.252.1] [5001] [4]......

connecting server 192.168.252.1 5001 4......

Prapare to Retry Connect Server[10s]

ENTER the offline MODE

离线时长 time\_out=[259200] offhour=[72] offmin=[0] 离线时间判定 off\_time=[1595471594] curr\_time=[1595471804] time\_out=[259200] recodr\_time=[1595471790]

OfflineTimeManagement return ret:0

日志解读：

关键字1：connecting server [IP] [Port] [X]

解 读1：链接服务器 地址和端口，X=4 链接异常 X=7 链接正常

关键字2：Prapare to Retry Connect Server

解 读2：超时重连

### 4.1.4 策略更新

* **终端重启更新策略**

**日志片段：**

Now restart trld service

Close the remete Client FD from remote\_recv\_watcher

Close the local SERVER FD

Terminal Version V6.0 - 6.00.191111

Server Version V6.0 - 6.00.200720.SC

read ip: 192.168.7.234

read port: 5001

connecting server [192.168.7.234] [5001] [4]......

connecting server 192.168.7.234 5001 4......

getpeername succeed...

getsockopt succeed...err=0!!!

ComputerName=[ti-deep-PC]

send checklogin frame[len=172]......

SendCheckLoginMsg Succeed!!!

EXIT Offline Mode

进入HandleResponse函数

HandleResponse 0x0186

byAutoLogin=[1]

Conect the server 192.168.7.234:5002

the policy is updated!

Now ExecAutoLogin [14625]!!!!!!

Terminal Version V6.0 - 6.00.191111

Server Version V6.0 - 6.00.200720.SC

UpdateWSPolicy2Kernel return 0

LoadPolicy2Kernel return 0

UpdatePdaPolicy2Kerne return 0

UpdateDmPolicy2Kernel return 0

UpdateGPrivPolicy2Kernel return 0

更新代理配置信息

操作内核驱动成功

HandleResponseCheckLogin::retry check login

Prapare to Retry Connect Server[10s]

connecting server [192.168.7.234] [5001] [4]......

connecting server 192.168.7.234 5001 4......

getpeername succeed...

getsockopt succeed...err=0!!!

ComputerName=[ti-deep-PC]

send checklogin frame[len=172]......

SendCheckLoginMsg Succeed!!!

EXIT Offline Mode

进入HandleResponse函数

HandleResponse 0x0186

byAutoLogin=[1]

Conect the server 192.168.7.234:5002

the policy is updated!

HandleResponseCheckLogin::A user has logged in [7][14625]!!!!!

Terminal Version V6.0 - 6.00.191111

Server Version V6.0 - 6.00.200720.SC

HandleResponseCheckLogin::开始更新驱动策略.... [7][14625]!!!!!

HandleResponseCheckLogin::更新工作目录策略成功 [0]!!!!!

HandleResponseCheckLogin::更新加解密驱动策略成功 [7][14625]!!!!!

HandleResponseCheckLogin::更新受控进程高级策略成功 [7][14625]!!!!!

HandleResponseCheckLogin::更新桌管驱动策略成功 [7][14625]!!!!!

HandleResponseCheckLogin::更新分组权限驱动策略成功 [7][14625]!!!!!

HandleResponseCheckLogin::IsValidTermMod [7][14625]!!!!!

HandleResponseCheckLogin::StartWork2Kernel [7][14625]!!!!!

HandleResponseCheckLogin::更新驱动策略成功.... [7][14625]!!!!!

* **服务器策略修改更新策略**

**日志片段：**

ManualUpdateRemotePolicy::开始更新驱动策略.... [7][14625]!!!!!

ManualUpdateRemotePolicy::重置本地策略时间.... [7][14625]!!!!!

Conect the server 192.168.7.234:5002

开始下载策略

ret = 1024, size = 1024

… …

ret = 232, size = 127208

接收策略时间:0

下载策略完成

policy file write done!

ImportTipFile\_result返回值:0

ManualUpdateRemotePolicy::更新工作目录策略成功 [7][14625]!!!!!

ManualUpdateRemotePolicy::更新加解密驱动策略成功 [7][14625]!!!!!

ManualUpdateRemotePolicy::更新受控进程高级策略成功 [7][14625]!!!!!

ManualUpdateRemotePolicy::更新桌管驱动策略成功 [7][14625]!!!!!

ManualUpdateRemotePolicy::更新分组权限驱动策略成功 [7][14625]!!!!!

ManualUpdateRemotePolicy::IsValidTermMod [7][14625]!!!!!

ManualUpdateRemotePolicy::StartWork2Kernel [7][14625]!!!!!

ManualUpdateRemotePolicy::更新驱动策略成功.... [7][14625]!!!!!

日志解读

关键字：size = 127208

解读：大小，特指策略包接受大小，最终和采集服务器中的Linux.tip文件大小一致

## 4.2 trld.log

trld.log常用于linux问题排查，比如追踪打开加密文件的过程，什么进程可以打开加密文件、什么进程不能打开加密文件。trld.log默认不产生，要产生该日志，必须先设置日志等级，然后重现问题，再查看。

设置日志等级命令：TRLD --log --level [n]

权限：无

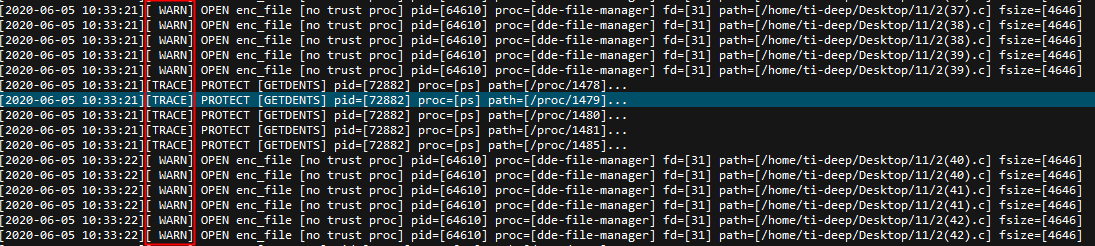
描述：设置日志等级（0-3：调试、警告、运行、错误）

主要是获取错误日志（即非受控程序打开加密文件产生的日志），可以TRLD --log --level 1或TRLD --log --level 0

其他日志研发人员使用较多。

非受控程序打开加密文件的关键字 no tust proc

日志片段：



日志解读：

[WARN] //日志信息等级，WARN是警告，非受控程序打开加密文件就是警告等级的

OPEN enc\_file //表示本日志是某进程打开加密文件

[no trust proc] //表示该程序是非受控程序

pid[64610] //进程PID

proc=[dde-file-manager] //进程名称

path=[/home/ti-deep/Desktop/11/2(37).c] //文件路径

fzize=[4646] //文件大小

## 4.3 获取受控程序md5值

获取md5值的目录是为了实现进程防伪冒，即防止其他进程伪冒成受控程序并通过网络通道等方式传输出明文。

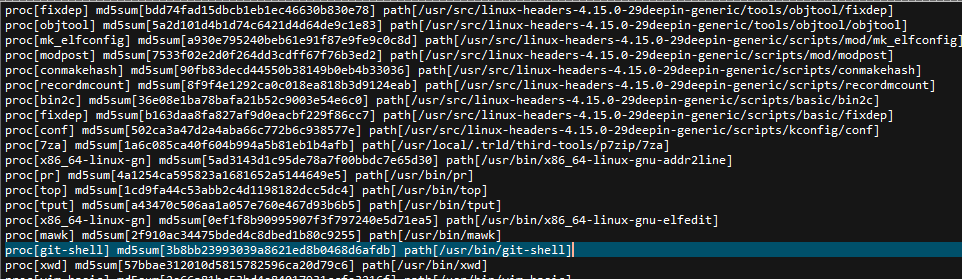
要获取md5值，首先必须确保linux受控程序已配置并已成功下发策略。

TRLD --collect-md5 start //开启获取md5值

TRLD --export-md5 /home/ten/Desktop/11.txt //导出md5值到文件

TRLD --collect-md5 stop //停止获取md5值

导出的md5值文件信息：

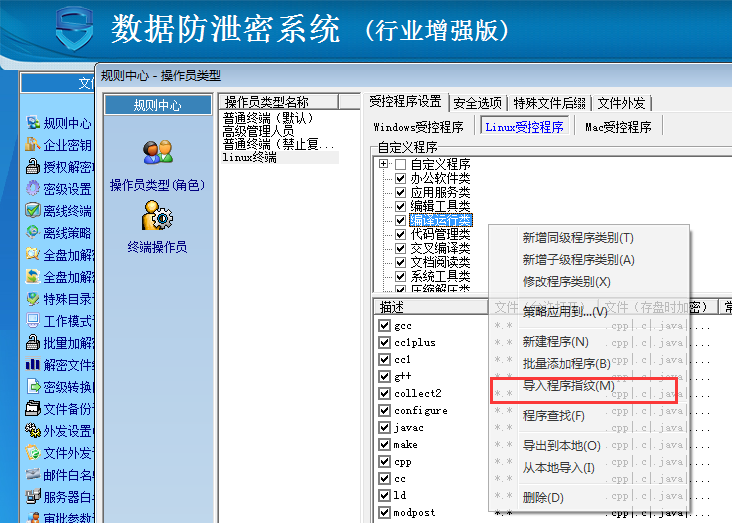


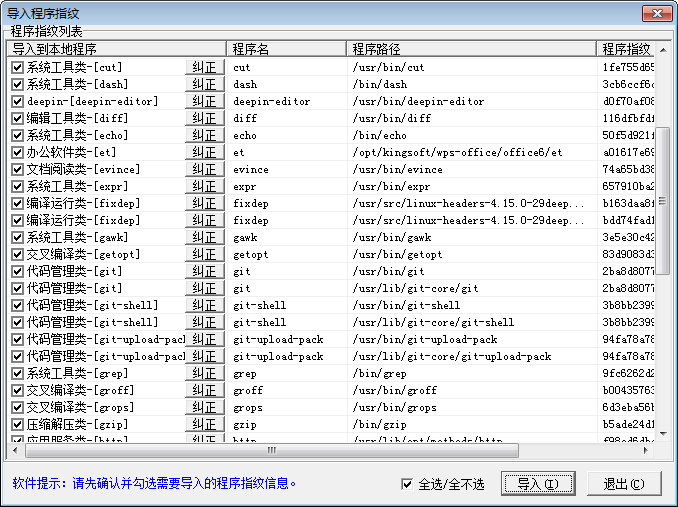
proc[XXXXX] //受控程序名称

md5sum[xxxx] //该程序的MD5值，32位16进制数

path[xxxx] //该程序的文件路径

使用该md5值文件：





# 5 macOS日志

## 5.1 trservice.log

trservice.log是记录macOS终端服务运行状态的日志，日志文件路径是/var/log/trld.log

常用日志信息：

### 5.1.1 没有分配加密模块，继而驱动停止工作

关键字 can not find computer\_id in Module\_Computer.db!!!!!

关键字 GetDriverWorkStat kennel don't work

日志举例：

send Busy singnal to local client [6]

send Busy singnal to local client [6]

启动快速加密组件...

启动备份进程...

can not find computer\_id in Module\_Computer.db!!!!!

GetDriverWorkStat kennel don't work

can not find computer\_id in Module\_Computer.db!!!!!

GetDriverWorkStat kennel don't work

can not find computer\_id in Module\_Computer.db!!!!!

GetDriverWorkStat kennel don't work

### 5.1.2 导入离线离线策略失败

关键字 导入永久离线策略失败：ret = -1

日志举例：

启动快速加密组件...

启动备份进程...

send Busy singnal to local client [6]

fopen MOD\_PATH failed

GetDriverWorkStat kennel don't work

send Busy singnal to local client [6]

fopen MOD\_PATH failed

GetDriverWorkStat kennel don't work

send Busy singnal to local client [6]

send Busy singnal to local client [6]

启动快速加密组件...

启动备份进程...

fopen MOD\_PATH failed

GetDriverWorkStat kennel don't work

send Busy singnal to local client [6]

Open tip file error[No such file or directory]

导入永久离线策略失败：ret = -1

日志分析，主要看 ret 后的值：

-1 //策略文件打不开

-2 //策略类型不是tip

-3 //不是macOS的策略

-7 //解析出错

（1）策略更新过程

**日志举例：**

send checklogin frame......

send Busy singnal to local client [7]

进入HandleResponse函数

case 0x01

prepare update policy!

PrapareUpdatePolicyOnline [10s]

进入HandleResponse函数

case 0x86

case 0x96

byAutoLogin=[1]

Conect the server 192.168.7.234:5002

短链接端口连接成功 sock\_fd=7

解析策略获取文件大小 file\_size = 124583

Policy Time=[1d66979ae640c80]

开始下载策略

ret = 2484, size = 2484

。。。

ret = 679, size = 124583

接收策略时间:0

下载策略完成

policy file write done!

打开策略文件[/tmp/macOS.tip]

提取策略文件[/tmp/vY4mHN]

mv [mv /usr/local/trld/data/temp/\* /usr/local/trld/data/]

rm /usr/local/trld/data/temp

ImportTipFile\_result返回值:0

HandleResponseCheckLogin::A user has logged in [8][4665]!!!!!

fopen TUR\_PATH failed

SendUpdateMsg2Trdm Success!!!!!!

**日志分析：**

1、Conect the server 192.168.7.234:5002

提示连接的服务器地址及端口（日志Conect单词拼写有误，忽略，如要查询关键字，还是用Conect）

2、解析策略获取文件大小 file\_size = 124583 ret = 679, size = 124583

主要包含检测到策略包的大小和策略策略下载进度，最后收到大小需要一致，且和

需要和采集目录下的 macOS.tip文件的实际大小一致

**（2）在线判断**

关键字 keepOnlineAlive:reccv=[OK!][[8]

有出现该日志信息，说明终端在线

## 5.2 trld.log

trld.log常用于macOS问题排查，比如追踪打开加密文件的过程，什么进程可以打开加密文件、什么进程不能打开加密文件。trld.log默认不产生，要产生该日志，必须先开启调试模式，然后重现问题，再查看。

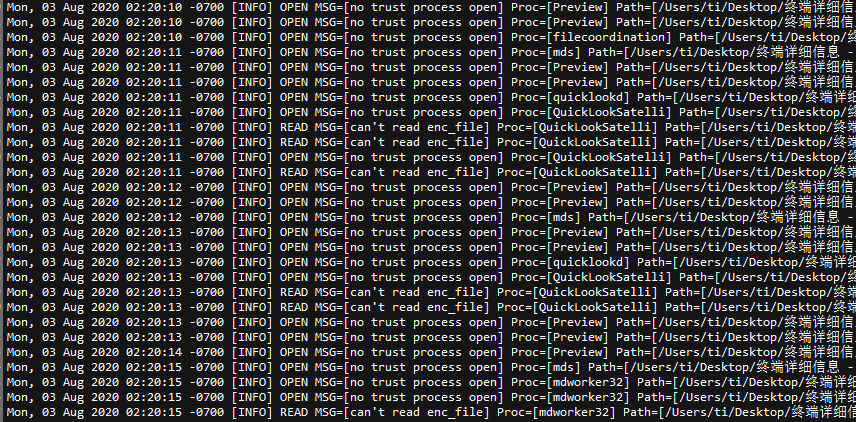
开启调试模式命令：TRLogin -d start

关闭调试模式命令：TRLogin -d stop

**主要是获取错误日志（即非受控程序打开加密文件产生的日志）。**

非受控程序打开加密文件的关键字 no tust process open

日志片段：



OPEN MSG=[no trust process open] //表示非受控程序打开加密文件

Proc //进程名称

Path //文件路径

READ MSG=[can’t read enc\_file] //非受控程序读取加密文件，结果是无法读取

# 6 审批日志

## 6.1 集成审批日志

LdApproval.log

**关键字：Error、Fail**

注：审批日志一般是失败才查看，日志查询Error或Fail可以定位到错误行，再看具体错误码或其他信息

正常的整个审批过程日志示例（为演示方便，以本机向本机申请解密为例）：

[2020-08-18 17:25:07]==>Approval.DataArrivedEx MID=<101>BT=<1>DT=<1>

//新建审批、收到审批或收到审批应答的标识，此处为申请者的审批组件收到新建审批的请求

[2020-08-18 17:25:08]GetUserInfo DataServerId<33>UserId<1>ComputerId<11539>GroupId<1>UserNo<1011>UserName<User11539>MakeType<0>

[2020-08-18 17:25:08]DecryptWaterClass.Destroy...

[2020-08-18 17:25:08]Get PenetratrEnDecryption file ext:[]

[2020-08-18 17:25:08]Need Judge UserRights! //是否需要判断文件密级

[2020-08-18 17:25:08]DecryptFrm IsWordFile<C:\Users\ten\Desktop\test33.xlsx>

[2020-08-18 17:25:08]DecryptWaterClass.Create...

[2020-08-18 17:25:08]GetWatermark:...

[2020-08-18 17:25:08]DecryptFrm IsWordFile<test33.xlsx>

[2020-08-18 17:25:09]Send TermUserReq To DstComputerId<1> iPos<11>

[2020-08-18 17:25:09]Send TermUserReq UserId<1>

[2020-08-18 17:25:09]==>OffLine.DataArrivedEx MID=<97>BT=<2>DT=<4>

[2020-08-18 17:25:09]UploadFile ServerId<33> ServerFilePath<$DATAPATH$\DataFile\2020-08-18\ApprovalUpLoad\{973BA0BD-6BA2-479E-B568-5C9801FC58BF}> LocalFilePath<C:\Users\ten\Desktop\test33.xlsx> By Server //申请的文件原始路径是C:\Users\ten\Desktop\test33.xlsx，将上传到服务器当天的ApprovalUpLoad目录下

[2020-08-18 17:25:10]DecryptFrm.SaveInfoToDbByItem ApprovalDecryptLog SUCC {973BA0BD-6BA2-479E-B568-5C9801FC58BF}

[2020-08-18 17:25:10]TermUserResp Count=<1> Info<1\_11539>

[2020-08-18 17:25:10]AnswerType:0 dwDstComputerId:65535 IfAutoApproval:false //不是系统自动应答流程

[2020-08-18 17:25:10]Send DecryptFileReq To DstComputerId<66> iPos<474> GUID<{973BA0BD-6BA2-479E-B568-5C9801FC58BF}>

[2020-08-18 17:25:10]Send DecryptFileReq To DstComputerId<0> iPos<474> GUID<{973BA0BD-6BA2-479E-B568-5C9801FC58BF}>

[2020-08-18 17:25:10]==>Approval.DataArrivedEx MID=<101>BT=<1>DT=<3>

//审批者的审批组件收到解密审批请求

[2020-08-18 17:25:10]InsertFileReport FilePath=C:\Inetpub\ftproot\Tipray\LdTerm\Repository\2020-08-18\DataFile\ApprovalUpLoad\{973BA0BD-6BA2-479E-B568-5C9801FC58BF} strSql=Update ApprovalDecryptLog Set SavePath=%s Where InfoId='{973BA0BD-6BA2-479E-B568-5C9801FC58BF}' InfoId={973BA0BD-6BA2-479E-B568-5C9801FC58BF} Success

//申请者日志提示审批日志已上传到服务器成功

[2020-08-18 17:25:11]收到解密申请:{973BA0BD-6BA2-479E-B568-5C9801FC58BF} //审批者收到解密申请

[2020-08-18 17:25:12]DecryptWaterClass.Destroy...

[2020-08-18 17:25:12]DecryptWaterClass.Create...

[2020-08-18 17:25:13]DecryptInfoFrm GetWatermark Doc WaterMark list Empty!

[2020-08-18 17:25:13]TDecryptInfoFrm.GetNeedDealApproval(33) Begin------->

[2020-08-18 17:25:13]TDecryptInfoFrm.GetNeedDealApproval Load RecordCount<1>

[2020-08-18 17:25:13]------------->TDecryptInfoFrm.GetNeedDealApproval(33) End

[2020-08-18 17:25:13]DecryptFrm IsWordFile<test33.xlsx>

[2020-08-18 17:25:13]Send TermUserReq To DstComputerId<1> iPos<11>

[2020-08-18 17:25:13]Send TermUserReq UserId<1>

[2020-08-18 17:25:13]==>OffLine.DataArrivedEx MID=<97>BT=<2>DT=<4>

[2020-08-18 17:25:14]TermUserResp Count=<1> Info<1\_11539>

[2020-08-18 17:25:14]LdApproval Connect!

[2020-08-18 17:25:15]Send ApprovalStepDeal To DstComputerId<66> iPos<56> GUID<{973BA0BD-6BA2-479E-B568-5C9801FC58BF}>

[2020-08-18 17:25:15]Send ApprovalStepDeal To DstComputerId<0> iPos<56> GUID<{973BA0BD-6BA2-479E-B568-5C9801FC58BF}> //审批者处理请求，可以是申请或拒绝

[2020-08-18 17:25:15]==>Approval.DataArrivedEx MID=<101>BT=<2>DT=<2>

//申请者的审批组件收到审批应答信息

[2020-08-18 17:25:15]Send DecryptFileResp To DstComputerId<0> iPos<167> GUID<{973BA0BD-6BA2-479E-B568-5C9801FC58BF}>

[2020-08-18 17:25:15]Send ApprovalFinish To DstComputerId<66> iPos<148> GUID<{973BA0BD-6BA2-479E-B568-5C9801FC58BF}> Type<1>

[2020-08-18 17:25:15]==>Approval.DataArrivedEx MID=<101>BT=<1>DT=<131>

[2020-08-18 17:25:16]ApprovalStepHide Recv<{973BA0BD-6BA2-479E-B568-5C9801FC58BF}>

[2020-08-18 17:25:16]收到解密申请应答:{973BA0BD-6BA2-479E-B568-5C9801FC58BF}

//申请者收到解密申请应答

[2020-08-18 17:25:18]DecryptWaterClass.Destroy...

[2020-08-18 17:25:18]DecryptWaterClass.Create...

[2020-08-18 17:25:18]DecryptInfoFrm GetWatermark Doc WaterMark list Empty!

[2020-08-18 17:25:18]TDecryptApplyInfoFrm.GetMyApplyApproval(1) Begin------->

[2020-08-18 17:25:18]TDecryptApplyInfoFrm.GetMyApplyApproval(1) RecordCount<1>

[2020-08-18 17:25:18]------->TDecryptApplyInfoFrm.GetMyApplyApproval(1) End

[2020-08-18 17:25:20]DecryptFrm IsWordFile<test33.xlsx>

[2020-08-18 17:25:20]Get PenetratrEnDecryption file ext:[]

[2020-08-18 17:25:21]Need Judge UserRights!

[2020-08-18 17:25:21]DecryptFile Src:<C:\Inetpub\ftproot\Tipray\LdTerm\Repository\2020-08-18\DataFile\ApprovalUpLoad\{973BA0BD-6BA2-479E-B568-5C9801FC58BF}>;Dest:<C:\解密文件\test33.xlsx>;Arlib ErrorCode<0>(0-Suc). //解密成功

[2020-08-18 17:25:21]PenetratrEnDecryption.IfCompressFile<C:\解密文件\test33.xlsx> False. //不需要压缩文件

[2020-08-18 17:25:21]DecryptFile Success SrcFile<C:\Inetpub\ftproot\Tipray\LdTerm\Repository\2020-08-18\DataFile\ApprovalUpLoad\{973BA0BD-6BA2-479E-B568-5C9801FC58BF}> //解密成功

### 6.1.1 申请解密失败错误码

|  |  |  |
| --- | --- | --- |
| **参数名** | **错误码** | **中文说明** |
| dsDecryptOK | 0 | //成功 |
| dsFileError | 1 | // |
| dsNotEncryptFile | 2 | // |
| dsPwdNotFound | 3 | // |
| dsPwdError | 4 | //终端保存的密钥信息与文件中保存的密钥信息不一致 |
| dsHeaderError | 5 | //一代加密文件头512解压失败 |
| dsNotSupport | 6 | //不支持的加密文件版本，超过当前已知的版本 |
| dsWriteError | 7 | //往加解密的临时文件中写入加解密后的内容失败 |
| dsReadError | 8 | // |
| dsMoveFileError1 | 9 | //是目标文件存在（比如把A文件解密到B文件，则B是目标文件，大多时候我们不指定把文件解密到另一个位置只是要替换原文件，那目标文件也是原文件），需要先将它改名，但改名失败了 |
| dsMoveFileError2 | 10 | //是删除改名后的原文件失败 |
| dsMoveFileError3 | 11 | //是把加解密出来的临时文件改名为目标文件失败 |

## 6.2 独立审批日志（待补充）

终端与独立审批通信返回的值（一般会在LdApprovalEx.log体现）：

-1: 与审批服务器断开连接！ //返回响应（GET与PUT请求）

200: 操作成功！ //返回响应（GET与PUT请求）

201: 操作成功！ //已经创建了实体，并在响应体中返回（POST请求）

204: 操作成功！ //已经删除了实体，因此没有返回的响应体（DELETE请求）

401: 操作失败！ //操作要求设置认证头。如果请求中有认证头，则提供的鉴证并不合法，或者用户未被授权进行该操作。

403: //操作被禁止，且不应重试。

404: //操作失败！请求的资源未找到。

405: //操作失败！使用的方法不能用于该资源。

409: //操作失败！该操作导致更新一个已被其他操作更新的资源，因此本更新不再有效。

415: //操作失败！请求体包含了不支持的媒体类型。

500: //操作失败！执行操作时发生了未知异常。

## 6.3 邮件白名单日志

### 6.3.1邮件白名单——配置日志

LdFileGate.log

如何查看邮件白名单配置信息

**关键字 CMailWhiteList**

例：CMailWhiteList::GetMailInfoFromDb(1)(864147081@qq.com) Success

说明该终端有分配了邮件白名单，[收件人白名单864147081@qq.com](mailto:收件人白名单864147081@qq.com)。

#### 6.3.1.1 查看发送加密附件给收件人白名单情况

**关键字 Send Mail或SenMail**

例：SendMail To<864147081@qq.com> is WhiteList,answer //收件人邮箱是白名单

Send Mail Has attachment<"邮件白名单日志详解.doc"> Decrypt MailDecryptType<0> EncodeType<0> EmlFile<0>! //发送附件《邮件白名单日志详解》是加密的，需解密

以下是一份发送加密附件给收件人白名单的完整日志信息分析：

##### 6.3.1.1.1 收件人是白名单情况

> Size<37> ProcessID<11672> ProcessName<C:\Foxmail 7.2\Foxmail.exe>

(2020-07-31 18:24:49)[10596]<20090>Proxy: connect to server success <113.96.232.106:25> socket=1636

(2020-07-31 18:24:50)[10596]From<From: "liujl@tipray.com" <liujl@tipray.com>

(2020-07-31 18:24:50)[10596]Find 250 11<250 Ok>

(2020-07-31 18:24:50)[10596]SendMail RCPT TO<RCPT TO: <864147081@qq.com>

> <m\_iRevCount: 0>

(2020-07-31 18:24:50)[10596]CrrntMail: <864147081@qq.com> DB Mail: <864147081@qq.com>

(2020-07-31 18:24:50)[10596]White Mail Address <864147081@qq.com> m\_iRevCount = 1!

(2020-07-31 18:24:50)[10596]MailOkFlag : 250 Ok

(2020-07-31 18:24:50)[10596]SendMail To<864147081@qq.com> is WhiteList,answer !

(2020-07-31 18:24:50)[10596]SendMail To1<864147081@qq.com>

(2020-07-31 18:24:50)[10596]bCanSend = False <RCPT TO: <864147081@qq.com>

(2020-07-31 18:24:50)[10596]Find DATA <1><RCPT TO: <864147081@qq.com>

><0>

(2020-07-31 18:24:50)[10596]Send Save Mail To<RCPT TO: <864147081@qq.com>

(2020-07-31 18:24:50)[10596]fopen<C:\Inetpub\ftproot\Tipray\LdTerm\log\NT AUTHORITY\_SYSTEM\smtp\1636\_20200731182450> success!

(2020-07-31 18:24:50)[10596]SendMail From<liujl@tipray.com> Not in WhiteList

(2020-07-31 18:24:50)[10596]fopen<C:\Inetpub\ftproot\Tipray\LdTerm\log\NT AUTHORITY\_SYSTEM\smtp\1636\_20200731182450> success!

(2020-07-31 18:24:51)[10596]fopen<C:\Inetpub\ftproot\Tipray\LdTerm\log\NT AUTHORITY\_SYSTEM\smtp\1636\_20200731182450> success!

(2020-07-31 18:24:51)[10596]SendMail Subject:<邮件白名单日志详解>< =?GB2312?B?08q8/rDXw/u1pcjV1r7P6r3i?=>

(2020-07-31 18:24:51)[10596]Find Content-Disposition: attachment

(2020-07-31 18:24:51)[10596]===========>Send Mail Has attachment<"邮件白名单日志详解.doc"> Decrypt MailDecryptType<0> EncodeType<0> EmlFile<0>!

(2020-07-31 18:24:51)[10596]Status = msFileTran

(2020-07-31 18:24:51)[10596]===========>Begin DecryptFile base64

(2020-07-31 18:24:51)[10596]It is 887D File!

(2020-07-31 18:24:51)[10596]fopen<C:\Inetpub\ftproot\Tipray\LdTerm\log\NT AUTHORITY\_SYSTEM\smtp\1636\_20200731182450> success!

(2020-07-31 18:24:51)[10596]fopen<C:\Inetpub\ftproot\Tipray\LdTerm\log\NT AUTHORITY\_SYSTEM\smtp\1636\_20200731182450> success!

(2020-07-31 18:24:57)[10596]!!!gaoyl debug

(2020-07-31 18:24:57)[10596]Send at .

(2020-07-31 18:24:57)[10596]SMTP OVER<.

> msEnd<5> BufPos<2293> BufLen<2294>!

(2020-07-31 18:24:57)[10596]Smtp Over Init

(2020-07-31 18:24:57)[10596] Send at QUIT

(2020-07-31 18:24:57)[10596]Send Mail Over!

##### 6.3.1.1.2 收件人是非白名单情况

> Size<37> ProcessID<12232> ProcessName<C:\Foxmail 7.2\Foxmail.exe>

(2020-07-31 19:27:27)[10596]From<From: "liujl@tipray.com" <liujl@tipray.com>

>

(2020-07-31 19:27:27)[10596]SendMail From<liujl@tipray.com> Not in WhiteList

(2020-07-31 19:27:27)[10596]SendMail Subject:<邮件白名单日志详解>< =?GB2312?B?08q8/rDXw/u1pcjV1r7P6r3i?=

>

(2020-07-31 19:27:27)[10596]Find Content-Disposition: attachment

(2020-07-31 19:27:27)[10596]===========>Send Mail Has attachment<"邮件白名单日志详解.doc"> Not Decrypt MailDecryptType<0> EncodeType<0> EmlFile<0>!

(2020-07-31 19:27:27)[10596]Status = msFileTran

(2020-07-31 19:27:27)[10596]===========>Begin DecryptFile base64

(2020-07-31 19:27:27)[10596]It is 887D File!

(2020-07-31 19:27:27)[10596]fopen<C:\Inetpub\ftproot\Tipray\LdTerm\log\NT AUTHORITY\_SYSTEM\smtp\1844\_20200731192727> success!

(2020-07-31 19:27:27)[10596]fopen<C:\Inetpub\ftproot\Tipray\LdTerm\log\NT AUTHORITY\_SYSTEM\smtp\1844\_20200731192727> success!

(2020-07-31 19:27:28)[10596]!!!gaoyl debug

(2020-07-31 19:27:28)[10596]Send at .

(2020-07-31 19:27:28)[10596]SMTP OVER<.

> msEnd<5> BufPos<2289> BufLen<2290>!

(2020-07-31 19:27:28)[10596]Smtp Over Init

(2020-07-31 19:27:28)[10596]AddSavleFileNode find QUIT <1>

(2020-07-31 19:27:28)[10596]Send Mail Over!

(2020-07-31 19:27:28)[10596]DeleteFile<C:\Inetpub\ftproot\Tipray\LdTerm\log\NT AUTHORITY\_SYSTEM\smtp\1844\_20200731192727> On No Resend

(2020-07-31 19:27:28)[10596]fopen<C:\Inetpub\ftproot\Tipray\LdTerm\log\NT AUTHORITY\_SYSTEM\smtp\1844\_20200731192727> success!

(2020-07-31 19:27:28)[10596]SaveFileNode::EndSaveFile <0><C:\Inetpub\ftproot\Tipray\LdTerm\log\NT AUTHORITY\_SYSTEM\smtp\1844\_20200731192727>

(2020-07-31 19:27:29)[10596]CreateThread GlobeSaveTimerThread success!m\_hTimerThread<720> errorcode=<0>

(2020-07-31 19:27:29)[10596]Send at QUIT aaaaaaaaaaaa

(2020-07-31 19:27:29)[10596]Send at QUIT aaaaaaaaaaaa

(2020-07-31 19:27:29)[10596]Send mail at Quit Msg

(2020-07-31 19:27:29)[10596]AddSavleFileNode find QUIT <0>

(2020-07-31 19:27:29)[10596]Send at QUIT

(2020-07-31 19:27:29)[10596]Send Mail Over!

##### 6.3.1.1.3 收件人既有白名单也有非白名单用户

> Size<8> ProcessID<11964> ProcessName<C:\Foxmail 7.2\Foxmail.exe>

(2020-07-31 19:46:56)[10596]Find 250<250 Ok>

(2020-07-31 19:46:56)[10596]Find 250 11<250 Ok>

(2020-07-31 19:46:56)[10596]SendMail RCPT TO<RCPT TO: <864147081@qq.com>> <m\_iRevCount: 0>

(2020-07-31 19:46:56)[10596]CrrntMail: <864147081@qq.com> DB Mail: <864147081@qq.com>

(2020-07-31 19:46:56)[10596]White Mail Address <864147081@qq.com> m\_iRevCount = 1!

(2020-07-31 19:46:56)[10596]MailOkFlag : 250 Ok

(2020-07-31 19:46:56)[10596]SendMail To<864147081@qq.com> is WhiteList,answer !

(2020-07-31 19:46:56)[10596]SendMail To1<864147081@qq.com>

(2020-07-31 19:46:56)[10596]bCanSend = False <RCPT TO: <864147081@qq.com>

>

(2020-07-31 19:46:56)[10596]SendMail RCPT TO<RCPT TO: <461250586@qq.com>

> <m\_iRevCount: 1>

(2020-07-31 19:46:56)[10596]CrrntMail: <461250586@qq.com> DB Mail: <864147081@qq.com>

(2020-07-31 19:46:56)[10596]SendMail to<461250586@qq.com> The Dest Is Not In WhiteList!

(2020-07-31 19:46:56)[10596]From<From: "liujl@tipray.com" <liujl@tipray.com>

>

(2020-07-31 19:46:56)[10596]SendMail From<liujl@tipray.com> Not in WhiteList

(2020-07-31 19:46:56)[10596]SendMail Subject:<邮件白名单日志详解>< =?GB2312?B?08q8/rDXw/u1pcjV1r7P6r3i?=

>

(2020-07-31 19:46:56)[10596]Find Content-Disposition: attachment

(2020-07-31 19:46:56)[10596]===========>Send Mail Has attachment<"邮件白名单日志详解.doc"> Not Decrypt MailDecryptType<0> EncodeType<0> EmlFile<0>!

(2020-07-31 19:46:56)[10596]===========>Send Mail Has attachment<"服务器白名单日志详解.doc"> Not Decrypt

(2020-07-31 19:46:56)[10596]===========>It is DecryptFile So Not Need Decrypt

(2020-07-31 19:46:56)[10596]Status = msFileTran

(2020-07-31 19:46:56)[10596]===========>Begin DecryptFile base64

(2020-07-31 19:46:56)[10596]It is 887D File!

(2020-07-31 19:46:57)[10596]fopen<C:\Inetpub\ftproot\Tipray\LdTerm\log\NT AUTHORITY\_SYSTEM\smtp\1828\_20200731194656> success!

(2020-07-31 19:46:57)[10596]!!!gaoyl debug

(2020-07-31 19:46:57)[10596]Send at .

(2020-07-31 19:46:57)[10596]SMTP OVER<.

> msEnd<5> BufPos<2289> BufLen<2290>!

(2020-07-31 19:46:57)[10596]Smtp Over Init

(2020-07-31 19:46:57)[10596]AddSavleFileNode find QUIT <1>

(2020-07-31 19:46:57)[10596]Send Mail Over!

(2020-07-31 19:46:57)[10596]Add Send Mail<邮件白名单日志详解> FileNode IP<113.96.208.92> Port<25> File<C:\Inetpub\ftproot\Tipray\LdTerm\log\NT AUTHORITY\_SYSTEM\smtp\1828\_20200731194656> Socket<1828>

(2020-07-31 19:46:57)[10596]AddSendFileNode <insert into MailSendLst (State,SaveTime,SendTimes,SocketId,ServerIp,Port,Title,Filepath) values (0,'2020-07-31 19:46:57',4,1828,'113.96.208.92',25 ,'邮件白名单日志详解','C:\Inetpub\ftproot\Tipray\LdTerm\log\NT AUTHORITY\_SYSTEM\smtp\1828\_20200731194656');>

(2020-07-31 19:47:01)[10596]Send at QUIT aaaaaaaaaaaa

(2020-07-31 19:47:01)[10596]Send at QUIT aaaaaaaaaaaa

(2020-07-31 19:47:01)[10596]Send mail at Quit Msg

(2020-07-31 19:47:01)[10596]AddSavleFileNode find QUIT <0>

(2020-07-31 19:47:01)[10596]Send at QUIT

(2020-07-31 19:47:01)[10596]Send Mail Over!

(2020-07-31 19:47:01)[10596]DeleteFile<C:\Inetpub\ftproot\Tipray\LdTerm\log\NT AUTHORITY\_SYSTEM\smtp\1828\_20200731194701> On No Resend

(2020-07-31 19:47:01)[10596]fopen<C:\Inetpub\ftproot\Tipray\LdTerm\log\NT AUTHORITY\_SYSTEM\smtp\1828\_20200731194701> success!

> Size<9> ProcessID<11964> ProcessName<C:\Foxmail 7.2\Foxmail.exe>

(2020-07-31 19:47:01)[10596]<20090>Proxy: Close client socket <127.0.0.1:54424> socket=1836

(2020-07-31 19:47:01)[10596]CProxy::DealTcpNode CS[-1] SS[-1]

(2020-07-31 19:47:04)[10596]ReadResendNodeFromIniEx m\_SendList.AddTail <邮件白名单日志详解><C:\Inetpub\ftproot\Tipray\LdTerm\log\NT AUTHORITY\_SYSTEM\smtp\1828\_20200731194656><2020-07-31 19:46:57>

(2020-07-31 19:47:04)[10596]CSendMailFile::TimerThreadProc Count<1>

(2020-07-31 19:47:04)[10596]Send MailNode<邮件白名单日志详解> Server<113.96.208.92><25> MailFile<C:\Inetpub\ftproot\Tipray\LdTerm\log\NT AUTHORITY\_SYSTEM\smtp\1828\_20200731194656>

(2020-07-31 19:47:04)[10596]Begin Connect ServerEx..

(2020-07-31 19:47:04)[10596]CreateSocket 1812

(2020-07-31 19:47:04)[10596]CreateSocket Success..

(2020-07-31 19:47:04)[10596]Begin Connect Server..

(2020-07-31 19:47:04)[10596]connect success...

(2020-07-31 19:47:04)[10596]setsockopt success!

(2020-07-31 19:47:04)[10596]End Connect Server..

(2020-07-31 19:47:04)[10596]CheckResponseEx Recv [2][2][0]

(2020-07-31 19:47:04)[10596]End Connect Server1..

(2020-07-31 19:47:04)[10596]Begin SendMail Server<113.96.208.92><25> File<C:\Inetpub\ftproot\Tipray\LdTerm\log\NT AUTHORITY\_SYSTEM\smtp\1828\_20200731194656>

(2020-07-31 19:47:04)[10596]CheckResponseEx Recv [2][5][0]

(2020-07-31 19:47:04)[10596]CheckResponseEx Recv [3][3][4]

(2020-07-31 19:47:04)[10596]CheckResponseEx Recv [3][3][4]

(2020-07-31 19:47:04)[10596]CheckResponseEx Recv [2][3][5]

(2020-07-31 19:47:04)[10596]CheckResponseEx Recv [2][5][0]

(2020-07-31 19:47:04)[10596]SendMailNode <RCPT TO: <864147081@qq.com>

>

(2020-07-31 19:47:04)[10596]CheckResponseEx Recv [2][5][0]

(2020-07-31 19:47:04)[10596]CheckResponseEx Recv [3][5][4]

(2020-07-31 19:47:04)[10596]SendMailNode DATA ok

(2020-07-31 19:47:05)[10596]SendMailNode Find [.]

(2020-07-31 19:47:08)[10596]CheckResponseEx Recv [2][5][0]

(2020-07-31 19:47:08)[10596]SendMailNode Server<113.96.208.92><25> File<C:\Inetpub\ftproot\Tipray\LdTerm\log\NT AUTHORITY\_SYSTEM\smtp\1828\_20200731194656> Success On(.)

### 6.3.2 发件白名单发送加密附件情况

如何查看配置信息：

**关键字：GetMailFromSendWhiteList**

例：CMailWhiteList::GetMailFromSendWhiteList(liujl@tipray.com) Success

说明这个发件人白名单中有这个邮箱地址（liujl@tipray.com）

以下是发件人是白名单情况下发送加密附件给非收件人白名单的完整日志信息分析：

> Size<37> ProcessID<11484> ProcessName<C:\Foxmail 7.2\Foxmail.exe>

(2020-07-31 20:22:55)[6208]From<From: "liujl@tipray.com" <liujl@tipray.com>

>

(2020-07-31 20:22:55)[6208]CrrntSendMail: <liujl@tipray.com> DB Mail: <liujl@tipray.com>

(2020-07-31 20:22:55)[6208]SendMail From<liujl@tipray.com> in WhiteList

(2020-07-31 20:22:55)[6208]SendMail Subject:<邮件白名单日志详解>< =?GB2312?B?08q8/rDXw/u1pcjV1r7P6r3i?=

>

(2020-07-31 20:22:55)[6208]SendMail RCPT TO<RCPT TO: <461250586@qq.com>

> <m\_iRevCount: 0>

(2020-07-31 20:22:55)[6208]SendMail to<461250586@qq.com> The Dest Is Not In WhiteList!

(2020-07-31 20:22:55)[6208]Mail Recv<250 Ok

> Size<8>

(2020-07-31 20:22:55)[6208]Find Content-Disposition: attachment

(2020-07-31 20:22:55)[6208]===========>Send Mail Has attachment<"邮件白名单日志详解.doc"> Decrypt MailDecryptType<0> EncodeType<0> EmlFile<0>!

(2020-07-31 20:22:55)[6208]Status = msFileTran

(2020-07-31 20:22:55)[6208]===========>Begin DecryptFile base64

(2020-07-31 20:22:55)[6208]It is 887D File!

(2020-07-31 20:22:55)[6208]fopen<C:\Inetpub\ftproot\Tipray\LdTerm\log\NT AUTHORITY\_SYSTEM\smtp\1640\_20200731202254> success!

(2020-07-31 20:22:55)[6208]fopen<C:\Inetpub\ftproot\Tipray\LdTerm\log\NT AUTHORITY\_SYSTEM\smtp\1640\_20200731202254> success!

(2020-07-31 20:22:55)[6208]!!!gaoyl debug

(2020-07-31 20:22:55)[6208]Send at .

(2020-07-31 20:22:55)[6208]SMTP OVER<.

> msEnd<5> BufPos<2673> BufLen<2674>!

(2020-07-31 20:22:55)[6208]Smtp Over Init

(2020-07-31 20:22:55)[6208]AddSavleFileNode find QUIT <1>

(2020-07-31 20:22:55)[6208]Send Mail Over!

(2020-07-31 20:22:55)[6208]DeleteFile<C:\Inetpub\ftproot\Tipray\LdTerm\log\NT AUTHORITY\_SYSTEM\smtp\1640\_20200731202254> On No Resend

(2020-07-31 20:22:55)[6208]fopen<C:\Inetpub\ftproot\Tipray\LdTerm\log\NT AUTHORITY\_SYSTEM\smtp\1640\_20200731202254> success!

## 6.4 其他日志关键字

查看邮箱网关的转链信息是否成功

**关键字：ChangeLink Client或20090**

例：[20090] ChangeLink Client<113.96.200.115><25> Server<113.96.200.115><25> SourceID<11484> Success//说明smtp 25端口已经转链到我们网关，可以支持邮件白名单。

注意：如果没搜索到这个信息，说明邮箱设置端口有问题，需要核对下。

# **7 服务器白名单日志**

## **7.1 旧服务器白名单**

LdFileGate

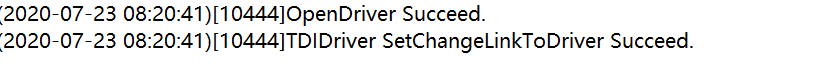
### 7.1.1 查看驱动是否正常

**关键字：TDIDriver**

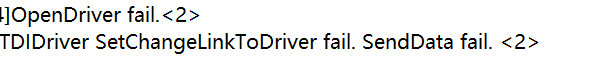
TDIDriver SetChangeLinkToDriver Succeed //说明服务器白名单所依赖的LdTDI驱动正常

TDIDriver SetChangeLinkToDriver fail. SendData fail. <2>//LdTDI驱动异常，2表示系统找不到指定的文件，意思是服务器白名单驱动不正常，这时候LdFileGate就没有相关的日志，发送邮件白名单或者上传附件，都不会有任何记录。

**驱动正常：**



**驱动异常：**

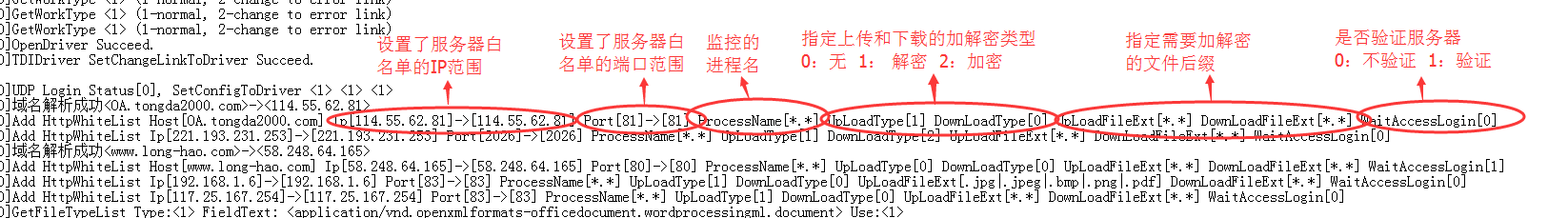


### 7.1.2 确认服务器白名单策略信息是否下发

**关键字：Add HttpWhiteList**

例如：

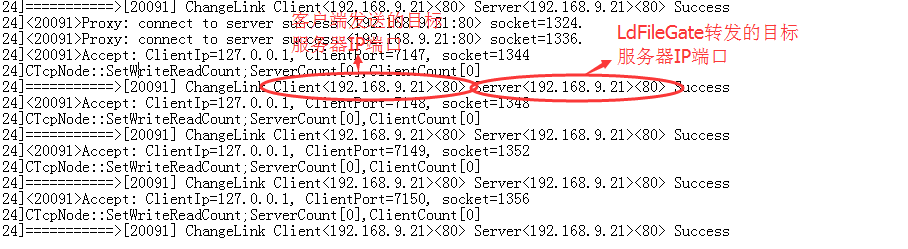
Add HttpWhiteList Ip[192.168.1.230]->[192.168.1.230] Port[80]->[80] ProcessName[\*.\*] UpLoadType[1] DownLoadType[2] UpLoadFileExt[\*.\*] DownLoadFileExt[\*.\*] WaitAccessLogin[0]



### 7.1.3 确认是否有转链信息

**关键字：ChangeLink Client**

如：===>[20091] ChangeLink Client<192.168.1.6><83> Server<192.168.1.6><83> SourceID<24500> Success



### 7.1.4 确认是否有解析到上传文件

**关键字：UpLoadFile Name**

例如：

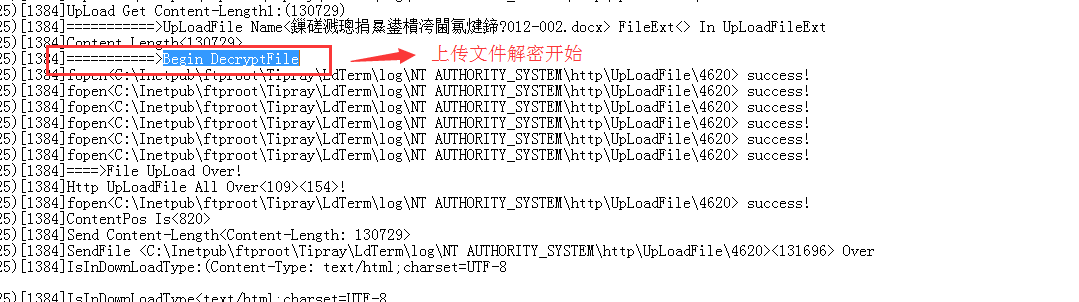
===========>UpLoadFile Name<FQ-H142-XL-001%2Epdf%2Epart0> FileExt<> In UpLoadFileExt



### 7.1.5 确认是否进入解密流程

**关键字：Begin DecryptFile**

例如：===========>Begin DecryptFile

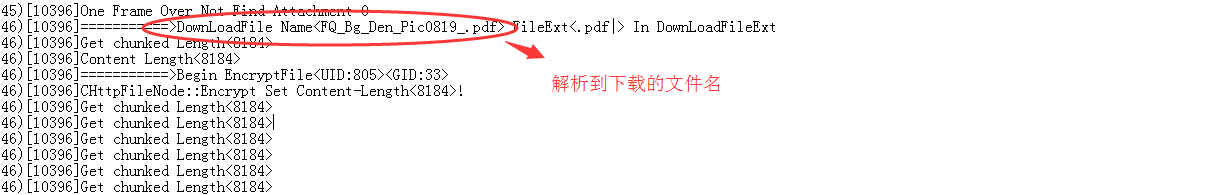


### 7.1.6 确认是否有解析到下载文件

**关键字：DownLoadFile Name**

例如：

===========>DownLoadFile Name<FQ\_Bg\_Den\_Pic0819\_.pdf> FileExt<.pdf|> In DownLoadFileExt



### 7.1.7 确认是否进入加密流程

**关键字：Begin EncryptFile**

例如：

===========>Begin EncryptFile<UID:805><GID:33>



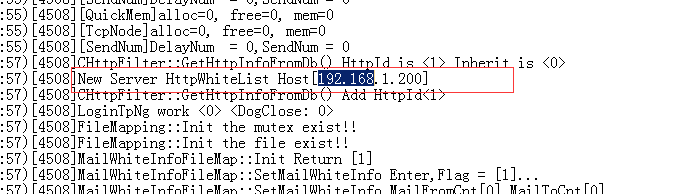
**注：旧服务器白名单是支持标准的且未进行gzip压缩的http协议**，经常会碰到不是标准的http协议，这时可以先用最新的LdFileGae.exe先试试，如果还是不行就要用fiddler等工具抓取数据包发给研发分析，有的经过研发修改LdFileGate.exe后可以支持。如果不支持，或者根本不是http协议，就要考虑用新服务器白名单，或者浏览器受控+桌管端口限制的方案了。

## **7.2 新服务器白名单**

LdNetMon.log、LdFileGate.log、ArDriver.log

### 7.2.1 LdFileGate.log

关键字 **New Server HttpWhiteList**可以查看新服务器白名单地址



### 7.2.2 LdNetMon.log

**关键字add setup white url**可以查看新服务器白名单地址（16进制转10进制）、端口、流量限制信息；

**white browser**可以查看该白名单允许的进程（通常是浏览器进程）

**例如：**

===========>add setup white url success<httpid=1><ip=c0a801c8 - c0a801c8><port=80 - 80><limit network = 0> <buffer msecond = 2000>. //第一条新服务器白名单<httpid=1>的地址是192.168.1.200，端口是80，流量限制是0（意思是访问白名单地址后，该进程再访问黑名单地址，允许的流量是0，也就是完全禁止访问）

set white browser name success<httpid=1><name=maxthon.exe><control=1>

// maxthon.exe是<httpid=1>的白名单进程

Set white browser name success<httpid=1><name=qqbrowser.exe><control=1>

//qqbrowser.exe是<httpid=1>的白名单进程

set white browser name success<httpid=1><name=theworld.exe><control=1>

// theworld.exe是<httpid=1>的白名单进程

set white browser name success<httpid=1><name=firefox.exe><control=1>

// firefox.exe是<httpid=1>的白名单进程

set white browser name success<httpid=1><name=chrome.exe><control=1>

// chrome.exe是<httpid=1>的白名单进程

set white browser name success<httpid=1><name=360se.exe><control=1>

// 360se.exe是<httpid=1>的白名单进程

set white browser name success<httpid=1><name=360chrome.exe><control=1>

//360chrome.exe是<httpid=1>的白名单进程

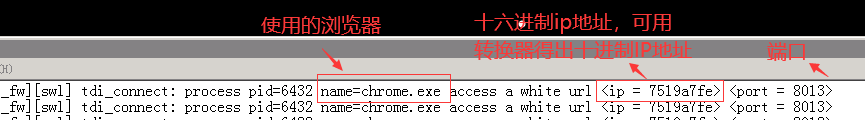


#### 7.2.2.1 访问白名单网站

**查看关键字：access a white url**

例如：

===========>tdi\_connect: process pid=6432 name=chrome.exe access a white url <ip = 7519a7fe> <port = 8013>

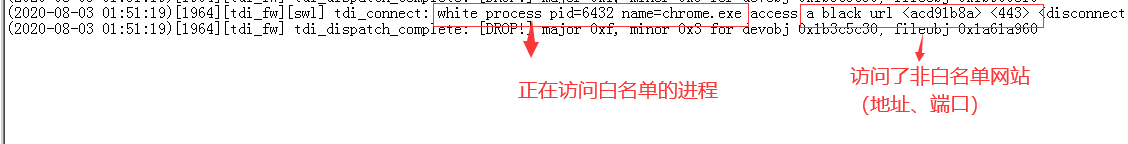


#### 7.2.2.2 开着白名单网站的浏览器访问非白名单网站被禁止

**查看关键字：access a black url**

例如：

===========>tdi\_connect: white process pid=6432 name=chrome.exe access a black url <acd91b8a> <443> <disconnect net>



非白名单进程访问白名单地址，查看关键字：black process

例如：

[tdi\_fw][swl] black process (pid=7896)telnet.exe access white url httpid=1 ip=c0a80106 port=82, forbit accesss!

### 7.2.3 ArDriver.log查看白名单设置和加解密过程

**关键字：SpEnc**

SpEnc\_DealRuleConfig R |\*.\*| //新服务器白名单策略，上传解密\*.\*

SpEnc\_DealRuleConfig W |.doc|.docx|.xls|.xlsx|.pdf //新服务器白名单策略，下载加密|.doc|.docx|.xls|.xlsx|.pdf

**上传解密过程：**

(2020-08-16 18:20:52)[3252]0000003252SpEnc\_PName\_AddSpEncProcess <chrome.exe>

(2020-08-16 18:20:52)[3252] 0000007716(chrome.exe)FsdCheckDirectoryFile Ret=0,Resion=0,Status=00000000,

<\Device\HarddiskVolume1\Users\ten\AppData\Local\Google\Chrome\User Data\Default\History>

//添加白名单进程名chrome.exe，添加白名单进程ID 7716，上面标红的0000007716很重要，表明PID 7716这个进程当前是白名单进程，然后才会有下面的日志：

(2020-08-16 18:40:08)[3252]0000007716[chrome.exe][FFFFF8A000CA4860][FFFFFA8004C38010]-1-Create.Fcb Exist,EncType=3,ReadDec=1,EncWrite=1,NeedEncCopy=0,EncCopy=0,SpecPath=0,DA=00120089,CD=00000001,OP=01000040

(2020-08-16 18:40:08)[3252]0000007716[chrome.exe][FFFFF8A000CA4860][FFFFFA8004963680]-2<00002ad9>File=\Device\HarddiskVolume1\Users\ten\Desktop\TEST1\nwtest.docx

(2020-08-16 18:40:08)[3252]0000000004[FFFFFA8004C38010][FFFFFA8003EA26E0]CanDecryptOnRead,ReadLen=10969,bytes=<50,4b,03,04><00,00,00,00>

//PID为7716的chrome.exe是可信程序，允许读加密文件C:\Users\ten\Desktop\TEST1\nwtest.docx

**下载加密过程：**

(2020-08-16 18:57:09)[3252]0000007716(chrome.exe)FsdCheckDirectoryFile

Ret=0,Resion=8,Status=00000000,<\Device\HarddiskVolume1\Users\ten\Downloads\nwtest.docx.crdownload>

(2020-08-16 18:57:09)[3252]0000007716[00001e24][chrome.exe]FileRenameInformation Set FCB\_STATE\_NEED\_ENCRYPT\_CLOSE <\Device\HarddiskVolume1\Users\ten\Downloads\nwtest.docx>

(2020-08-16 18:57:09)[3252]0000007716[chrome.exe](FFFFF8A003E82C70)ReportList\_AddReport:flag=5, network=0, iPos=5, pid=7716, file=\Device\HarddiskVolume1\Users\ten\Downloads\nwtest.docx

// FileRenameInformation Set FCB\_STATE\_NEED\_ENCRYPT\_CLOSE

// FileRenameInformation Set FCB\_STATE\_NEED\_ENCRYPT\_CLOSE是识别到白名单进程chrome.exe将下载的临时文件改名为nwtest.docx了，而.docx是在下载加密列表里的，所以后续上报应用层加密处理，即ReportList\_AddReport:flag=5

#### 7.2.3.1 附：驱动上报应用层加解密的编号列表

atEncAfterProcessClose = 2,//进程关闭后加密

atDecNow = 3,//马上解密

atDelNow = 4,//马上删除

atEncNow = 5,//马上加密

atUpdateEncVersionToV7 = 6, //转换成二代加密

atSmartEnc = 8, //需要智能加密分析进行加解密

# 8 应用服务器安全接入系统日志

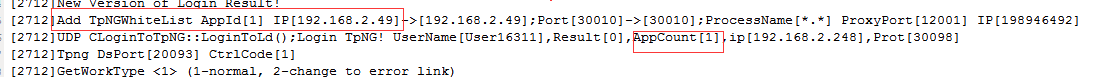
## 8.1 应用服务器安全接入系统日志——配置日志

LdFileGate.log

策略下发：

**关键字：Add TpNGWhiteList**

例：



Result[0] //说明终端准入已验证成功，操作员登录也验证成功

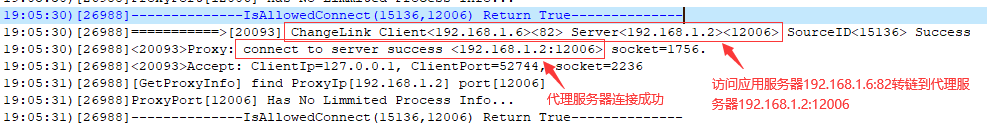
AppCount[1] //说明该终端已分配一个应用服务器

Ip[192.168.2.248],Port[30098] //应用接入系统服务器的IP、端口

Add TpNGWhiteList AppId[1] //第1个应用服务器IP、端口、进程（一般为\*.\*）、代理端口，以此类推如果有第2个应用服务器的话，就是AppId[2]

## 8.2 应用服务器安全接入系统日志——现象日志

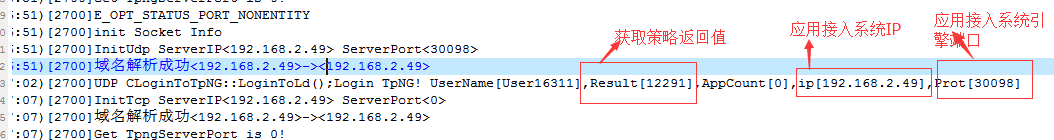
### 8.2.1 正常转链日志



ChangeLink Client //访问该地址、端口被转链

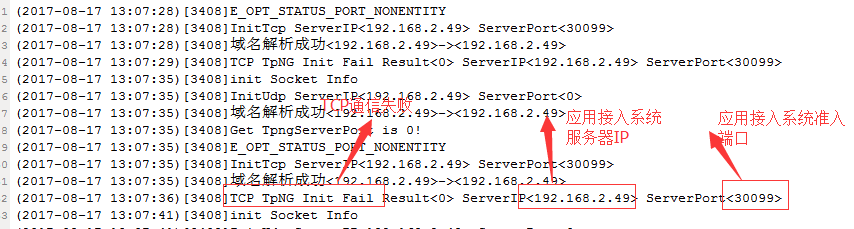
Server //转链到该代理服务器、端口

### 8.2.2 UDP错误日志



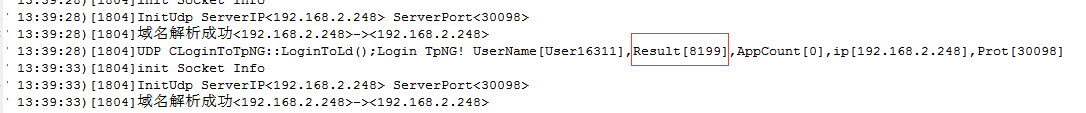
Result[12291] //Socket错误，说明通信失败

### 8.2.3 TCP错误日志



Tcp TpNG Init Fail //说明TCP通信失败

### 8.2.4 错误8199日志



Result[8199] //找不到指定终端

## 8.3 附：接入系统错误码

8193 打开数据库失败

8194 接收的是空内容

8195 等待审批

8196 审批不通过

8197 密码错误（用户登入或修改密码时）或缺少用户

8198 数据库操作失败，插入，修改，查询

8199 在准入设备表里查询不到此设备

8200 注册码无效

8201 数据不完整

8202 没有应用系统

8203 其他错误

8204 账号禁用

8205 账号未绑定此设备

8206 审批不通过

8207 此账号不存在

8208 此终端不用升级

8209 系统未注册

8210 系统终端数量超过

8211 注册时间超过

8212 狗不匹配

8213 没有狗的信息

8214 此用户没有绑定狗

12289 未登录

12290 未知错误

12291 socket错误

12292 找不取dll

## 8.4 应用接入系统macOS日志说明

### 8.4.1 查看代理是否运行

* 查看proxy是否运行 【ps -ef | grep proxy】
* 查看rinetd\_ex是否运行【ps -ef | grep rinetd\_ex】
* 如果安装过程没出现异常，正常情况下proxy跟rinetd\_ex进程会在运行，**缺一不可**

### 8.4.2 关键日志说明

|  |  |  |  |
| --- | --- | --- | --- |
| **组件** | **对应日志** | **关键字** | **说明** |
| proxy | /var/log/trproxy.log | login successfully | 终端登录应用接入成功 |
| get serverlist reply successfully | 获取应用服务器列表成功 |
| parse processInfo list successful | 解析进程控制信息成功 |
| proxy\_setting | /var/log/trproxy\_setting.log | proxy\_info\_count count | 配置的应用服务器个数 |
| rinetd\_ex | /var/log/rinetd\_ex.log | permit | 结合permit前的日志信息，可判定某个进程允许访问应用服务器 |
| not permit | 结合not permit前的日志信息，可判定某个进程不允许访问应用服务器 |

### 8.4.3 部分问题排查说明

#### 8.4.3.1 查看具体的转链信息

**查看配置文件/usr/local/trld/etc/server\_list\_info**，配置格式如下：

id\_0=7 ==> 应用服务器列表ID

ip\_0=192.168.2.156 ==> 目标应用服务器IP

port\_0=90 ==> 目标应用服务器Port

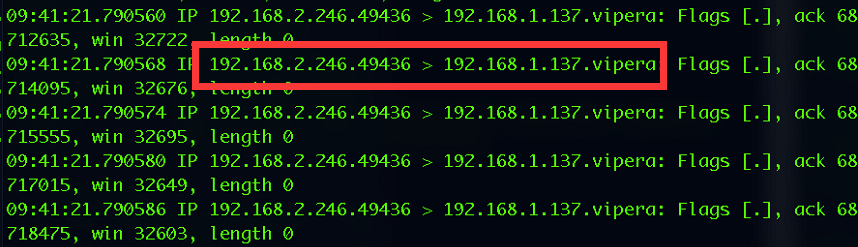
proxy\_ip\_0=192.168.1.137 ==> 代理服务器IP

proxy\_port\_0=12012 ==> 代理服务器端口

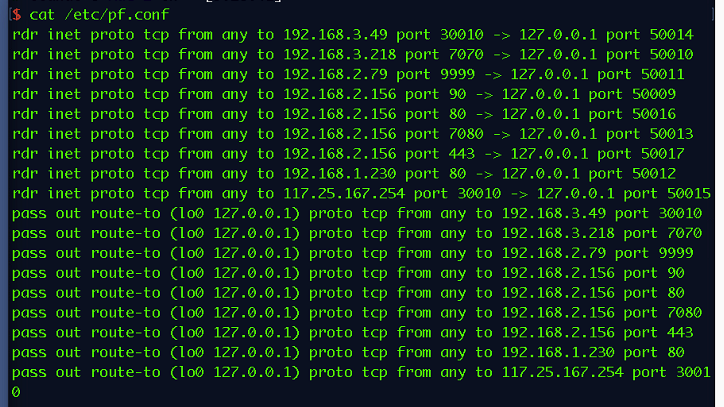
process\_list\_0=ab;7bef0ce3b106fa49afd84d3e548fea1b ==>允许访问此应用服务器的进程名及其md5值信息，多组信息使用‘|’隔开

#### 8.4.3.2 查看访问的数据是否有经过代理服务器转发

在终端上通过命令“sudo tcpdump host 代理服务器IP192.168.1.137，其次在终端上访问目标应用服务器，如果有明显的数据发送过程，且信息吻合，则说明访问的应用服务器是有经过代理服务器转发的。如下图：

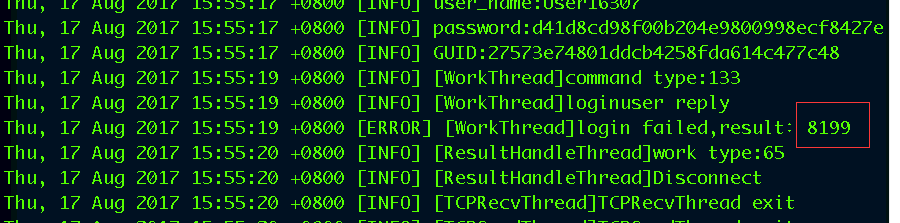


（1）查看MAC防火墙配置信息【cat /etc/pf.conf】

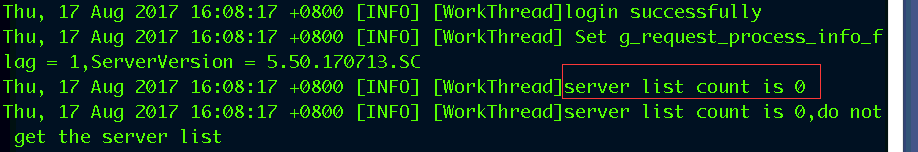


（2）错误日志说明

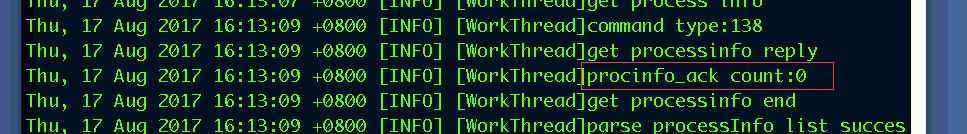
* 若访问失败，查看日志 cat /var/log/trproxy.log;Login failed,result: 8199,相应错误码和Windows错误码一样，按Windows排查方式排查



* 若日志提示server list count is 0,说明应用服务器未分配，排查方式如Windows



* 日志提示“procinfo\_ack count:0”,表示无配置进程绑定信息，全部放行



### 8.4.4 应用接入系统Linux日志说明

#### 8.4.4.1 查看代理是否运行、重启代理的方法

* 查看proxy是否运行 【ps -ef | grep proxy】
* 查看rinetd\_ex是否运行【ps -ef | grep rinetd\_ex】
* 如果安装过程没出现异常，正常情况下proxy跟rinetd\_ex进程会在运行，缺一不可
* 重启代理方法【sudo killall proxy; sudo /usr/local/trld/bin/proxy】
* 查看日志确认是否登录成功【/var/log/proxy.log中查看关键字“login successfully”】

#### 8.4.4.2 关键日志说明

|  |  |  |  |
| --- | --- | --- | --- |
| **组件** | **对应日志** | **关键字** | **说明** |
| proxy | /var/log/trproxy.log | login successfully | 终端登录应用接入成功 |
| get serverlist reply successfully | 获取应用服务器列表成功 |
| parse processInfo list successful | 解析进程控制信息成功 |
| proxy\_setting | /var/log/trproxy\_setting.log | proxy\_info\_count count | 配置的应用服务器个数 |
| rinetd\_ex | /var/log/rinetd\_ex.log | permit | 结合permit前的日志信息，可判定某个进程允许访问应用服务器 |
| not permit | 结合not permit前的日志信息，可判定某个进程不允许访问应用服务器 |

#### 8.4.4.3部分问题排查说明

（1）查看具体的转链信息

查看配置文件/usr/local/trld/etc/server\_list\_info，配置格式如下：

id\_0=7 ==> 应用服务器列表ID

ip\_0=192.168.2.156 ==> 目标应用服务器IP

port\_0=90 ==> 目标应用服务器Port

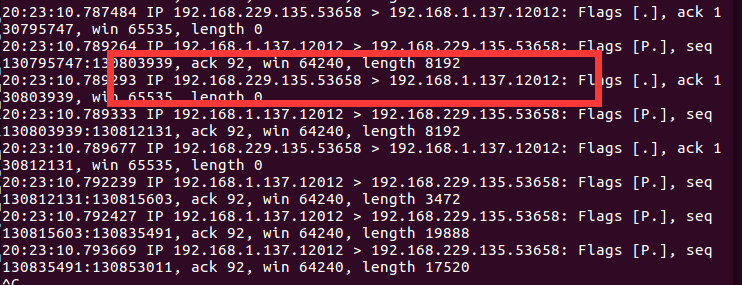
proxy\_ip\_0=192.168.1.137 ==> 代理服务器IP

proxy\_port\_0=12012 ==> 代理服务器端口

process\_list\_0=ab;7bef0ce3b106fa49afd84d3e548fea1b ==>允许访问此应用服务器的进程名及其md5值信息，多组信息使用‘|’隔开

（2）查看访问的数据是否有经过代理服务器转发【命令：sudo tcpdump host IP地址】

在终端上通过命令“sudo tcpdump host 代理服务器IP【如192.168.1.137】，其次在终端上访问目标应用服务器，如果有明显的数据发送接收过程，且信息吻合，则说明访问的应用服务器是有经过代理服务器转发的。如下图：

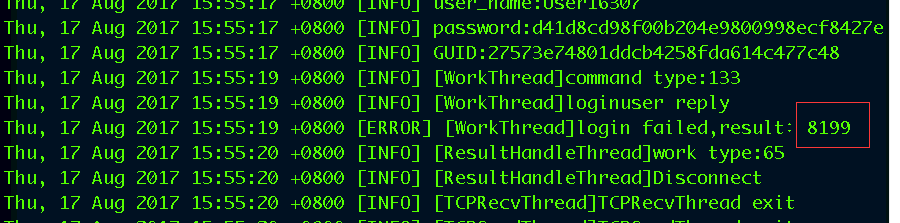


（3）查看Linux防火墙配置信息是否正常【命令：sudo iptables -n -t nat -L】

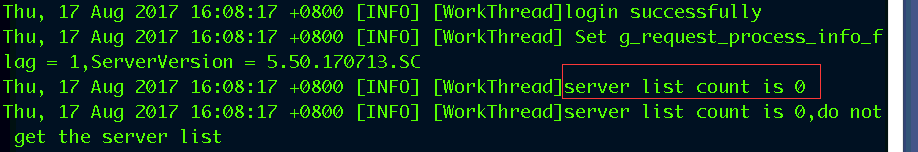


（4）错误日志说明

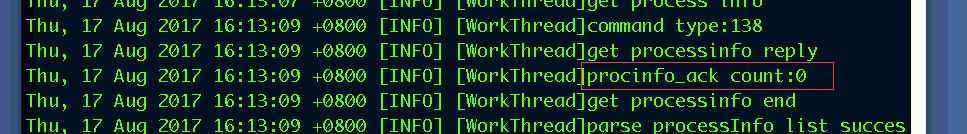
* 若访问失败，查看日志 cat /var/log/trproxy.log;Login failed,result: 8199, 相应错误码和Windows错误码一样，按Windows排查方式排查



* 若日志提示server list count is 0,说明应用服务器未分配，排查方式如Windows



* 日志提示“procinfo\_ack count:0”,表示无配置进程绑定信息，全部放行



# 9 离线策略日志

ArWorkStation.log

**搜索LoadOffline关键字**

[LdWs]DealLoadOfflineRep LoadOfflineFile fail

[1]错误值 -7，表示时间定义非法；

[2]错误值 -8，表示离线策略与导入的用户不匹配

## 9.1 离线策略日志--现象日志

### 9.1.1 提示“装载离线策略文件失败：时间定义非法”

(2014-01-21 10:27:46)[2444][LdWs]IsOverFile B = <2014-01-21 00:00:00>,E = <2014-02-11 23:59:59>,L = <2014-01-20 17:12:17>

(2014-01-21 10:27:46)[2444][LdWs]DealLoadOfflineRep LoadOfflineFile fail.<-7>

B ：开始时间；E：结束时间；L：当前时间；L的时间必须在B、E之间

### 9.1.1 提示“装载离线策略文件失败:该文件不允许被导入”

(2011-07-25 11:31:11)[5860][LdWs]DealLoadOfflineRep LoadOfflineFile fail.<-8>

该离线策略文件不属于该用户的

# 10 水印设置日志

## 10.1 屏幕水印设置

ArWorkstation.log

**关键字：ScreenWP\_Process或CScreenWPSetInfo**

例：

(2020-07-20 10:22:13)[16364]select ProcessName From ScreenWP\_Process where GroupId=65535

//针对整个组织架构（GroupId=65535是指所有分组，即整个公司）设置屏幕水印策略

(2020-07-20 10:22:13)[16364][Arws]CScreenWPSetInfo::ReadConfigInfoFromDbEx.<m\_dwProcessCount: 1>

//共有一个受控程序

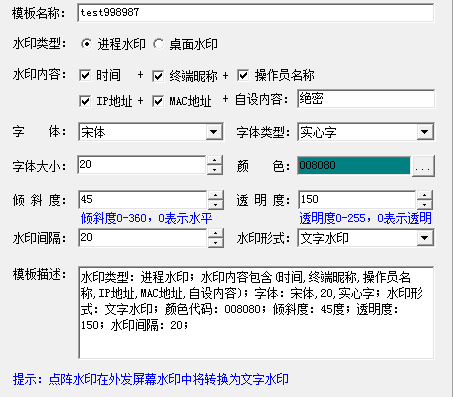
(2020-07-20 10:22:13)[16364][Arws]CScreenWPSetInfo::ReadConfigInfoFromDbEx.<ProcessName: notepad.exe>

//对记事本进程notepad.exe设置屏幕水印策略

(2020-07-20 10:22:13)[16364][Arws]CScreenWPSetInfo::ReadConfigInfoFromDb Success<IsWaterMark=1><FGorBG=0><绝密><dwEscapement=45><dwConstantAlpha=150><宋体><20><LINXY-PC><dwScreenMarkType:1><m\_dwProcessCount:1> <NickName:LINXY-PC>

//设置的具体屏幕水印策略，其中<FGorBG=0>表示前景（目前进程水印是支持前景，打印水印才有前景和背景之分），<绝密>自设内容为“绝密”，<m\_dwProcessCount:1>表示共有1个受控程序（如果=0，则为桌面水印，设置进程水印但未设置进程，和直接设置桌面水印是一样的效果），<NickName:LINXY-PC>表示要显示终端昵称，终端昵称为“LINXY-PC”

具体设置如下图所示：

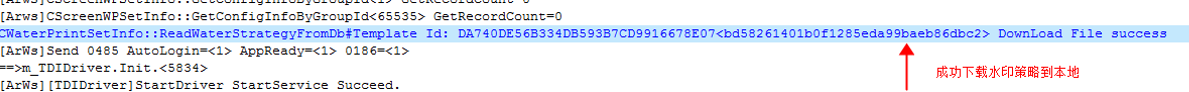


## 10.2 打印水印设置日志

ArWorkstation.log

**关键字：CWaterPrintSetInfo**

例：





## 10.3 外发水印设置日志

ldx.log

**关键字：Screen Water**

### 10.3.1 外发屏幕水印

**例：**

[2020-07-21 09:13:31][PID:13456][TID:9436]@@@ Begin Add Screen Water

[2020-07-21 09:13:31][PID:13456][TID:9436]@@@ Begin ScreenInitialize Screen Water:

Global\Ldx\_ShareMem\_13456

//外发屏幕水印开始标志

[2020-07-21 09:13:31][PID:13456][TID:9436]@@@ Dll Name:

C:\Users\linxy\AppData\Local\Temp\@tiprayldx@\693f3753f6de752c5f9429935e7968cf\LdWaterMarkHook32.dll hDll: 68280000

[2020-07-21 09:13:31][PID:13456][TID:9436]@@@ Print Dll Path:

C:\Users\linxy\AppData\Local\Temp\@tiprayldx@\693f3753f6de752c5f9429935e7968cf\LdPrintMonitor.dll Dll Handle: 05260000

[2020-07-21 09:13:31][PID:13456][TID:9436]@@@ End ScreenInitialize Screen Water

//外发屏幕水印结束标志

[2020-07-21 09:13:31][PID:13456][TID:9436]TScreenWaterMgr.CreateShareMem 获取共享内存映射地址成功

[2020-07-21 09:13:31][PID:13456][TID:9436]@@@ End Add Screen Water

### 10.3.2 外发打印水印

外发文件有设置屏幕水印并允许打印，则打印时就会带上一样的水印，日志比外发屏幕水印多了一条生成外发打印水印图片的。

**例：**

[2020-07-21 09:40:14][PID:14704][TID:6108]Ldx End Build Tree

[2020-07-21 09:40:16][PID:14704][TID:6108]@@@ Begin Add Screen Water

[2020-07-21 09:40:16][PID:14704][TID:6108]@@@ Begin ScreenInitialize Screen Water:

Global\Ldx\_ShareMem\_14704

//外发屏幕水印开始标志

[2020-07-21 09:40:16][PID:14704][TID:6108]@@@ Dll Name:

C:\Users\linxy\AppData\Local\Temp\@tiprayldx@\693f3753f6de752c5f9429935e7968cf\LdWaterMarkHook32.dll hDll: 68280000

[2020-07-21 09:40:16][PID:14704][TID:6108]@@@ Print Dll Path:

C:\Users\linxy\AppData\Local\Temp\@tiprayldx@\693f3753f6de752c5f9429935e7968cf\LdPrintMonitor.dll Dll Handle: 050B0000

[2020-07-21 09:40:16][PID:14704][TID:6108]@@@ End ScreenInitialize Screen Water

//外发屏幕水印结束标志

[2020-07-21 09:40:16][PID:14704][TID:6108]TScreenWaterMgr.CreateShareMem 获取共享内存映射地址成功

[2020-07-21 09:40:16][PID:14704][TID:6108]@@@ Copy Print Information: Type: 1

[2020-07-21 09:40:16][PID:14704][TID:6108]@@@ Image Info: Col: !2 Row: 2

[2020-07-21 09:40:16][PID:14704][TID:6108]@@@ Image Path Is:

C:\Users\linxy\AppData\Local\Temp\@tiprayldx@\693f3753f6de752c5f9429935e7968cf\Print\_Water\_Image.png

//生成外发打印水印图片

[2020-07-21 09:40:16][PID:14704][TID:6108]@@@ End Add Screen Water

## 10.4 word文档水印设置日志

ArWorkstation.log

**关键字：AddDocxWaterM或CaddDocxWaterM**

例：



# 11 上网限制日志

## 11.1 上网限制日志——配置日志

ArwCapture.log

### 11.1.2 端口限制

**关键字：PortList\_ReadFromDbTable或SysPortBlock\_BlockSome**

例： PortList\_ReadFromDbTable,SysPortBlock\_BlockSome: ExeName:360chrome.exe, BeginIp:fea71975, EndIp:fea71975, BeginPort:8800, EndPort:8800, Protocol:1,GroupId:0, UserId:10209, TimeId:1

表示10209这台终端禁止使用访问117.25.167.52:8800这个网站

**ExeName：\*.\***  表示所有进程

**BeginIp：fea71975** 表示要禁止访问的IP范围的起始IP(16进制),fea71975即75.19.a7.fe即117.25.167.254

**EndIp：fea71975** 表示要禁止访问的IP范围的结束IP(16进制),fea71975即75.19.a7.fe即117.25.167.254

**BeginPort:8800** 表示要禁止访问的端口范围的起始端口

**EndPort:8800** 表示要禁止访问的端口范围的结束端口

**Protocol:1** 表示禁用的传输协议，1代表TCP协议，2代表UDP协议

**TimeID：2** 表示时间规则，1代表默认时间安排表（全天候），2代表自定义时间安排表2（以此类推）

### 11.1.3 网站限制

禁止访问以下网页：

关键字：SysHttpBlock\_BlockSome

例： HttpList\_ReadFromDbTable: table:SysHttpBlock\_BlockSome,Url:taobao.com, TimeID:2

表示禁止访问url包含taobao.com的网站（淘宝网站）

只允许访问以下网页：

关键字：SysHttpBlock\_AllowSome

例： HttpList\_ReadFromDbTable: table:SysHttpBlock\_AllowSome,Url:dangdang.com, TimeID:1

表示只允许访问url包含dangdang.com的网站（当当网）

禁止访问所有网页：

关键字：SysPortBlock\_BlockSome

例： PortList\_ReadFromDbTable,SysPortBlock\_BlockSome: ExeName:\*.\*, BeginIp:0, EndIp:ffffffff, BeginPort:80, EndPort:80, Protocol:1,GroupId:0, UserId:10209, TimeId:1

例： PortList\_ReadFromDbTable,SysPortBlock\_BlockSome: ExeName:\*.\*, BeginIp:0, EndIp:ffffffff, BeginPort:443, EndPort:443, Protocol:1,GroupId:0, UserId:10209, TimeId:1

表示禁止任意程序访问80或443端口

80端口：为HTTP即超文本传输协议的默认端口，通常网址是通过http+地址+:端口号来访问网站，由于浏览http网页默认的端口号是80，如果一个http网址后面没有跟着:端口号，那么就是80端口。

443端口：为HTTPS即超文本传输协议的默认端口，通常网址是通过https+地址+:端口号来访问网站，由于浏览https网页默认的端口号是443，如果一个https网址后面没有跟着:端口号，那么就是443端口。

## 11.2 上网限制日志——现象日志

ArwCapture.log

用360极速浏览器访问访问117.25.167.254:8800这个网站后，看端口限制策略：

关键字：Check failed

例： BlockPort\_Check Failed, ProcessName:360chrome.exe, Ip:7519a7fe, Port:8800!

注：端口限制配置日志里显示的16进制地址是反着看的，即117.25.167.254是fea71975；而端口限制现象发生时显示的16地址是正着看的，即117.25.167.254是7519a7fe

用浏览器直接输入域名访问淘宝网站后，看网页限制策略：可以尝试：HttpsCheck OUName

关键字：HttpsCheck return FALSE

例： HttpsCheck return FALSE,url:taobao.com, szOUName:www.taobao.com

注：现在网站大部分是https的，如果是http网站，那么：

关键字：HttpCheck return FALSE

例： HttpsCheck return FALSE,url:taobao.com, szOUName:www.taobao.com

# 12 应用程序限制日志

## 12.1 应用程序限制日志——配置日志

ArWorkstation.log

### 12.1.1 应用程序限制

关键字：Block AddAppRecord

例：Block AddAppRecord <MSPAINT.EXE> <MSPAINT.EXE> <> TimeId = <1>

表示禁止使用画图程序

<MSPAINT.EXE> //表示被限制的程序

TimeID=<2> //表示时间规则，1代表默认时间安排表（全天候），2代表自定义时间安排表2（以此类推）

### 12.1.2 窗口标题限制

关键字：WinTitleBlock

注：5.52版本以后窗口标题限制没有设置日志了，只有当触发了窗口标题限制时，才会看到是匹配到了哪一条窗口标题限制，详见窗口标题限制现象日志

只是通过搜索WinTitleBlock可以查到这个功能有没有启动，以及设置或修改窗口标题限制时，有没有下发该数据库，例如：

### 12.1.3 禁止受控程序改名

关键字：EncryptType的CtrlCode

查询EncryptType行，看这一行的CtrlCode=多少，由十进制转换为二进制，看“禁止受控程序改名”这一位的值是0还是1【0表示不打钩，1表示打钩】

注：EncryptType行CtrlCode值表示的是操作员类型/安全选项/其他设置，以及操作员类型/高级设置的各个选项【0表示不打钩，1表示打钩】，二进制从右到左每一位对应的选项列表：

= 1<<0,    //禁止ole拖拽

= 1<<1,    //禁止删除文件

= 1<<2,    //允许制作打印外发文件

= 1<<3,    //自动同意解密

= 1<<4,    //自动同意外发

= 1<<5,    //表示打开网上邻居的文件不自动加密

= 1<<6,    //禁止虚拟打印到文件

= 1<<7,    //禁止应用程序改名

= 1<<8,    //允许explorer压缩文件

= 1<<9,    //进程隐藏

= 1<<10,  //关闭批量加密功能

= 1<<11,  //允许制作直接外发文件

= 1<<12,  //关闭批量加密记录

= 1<<13,  //关闭批量解密记录

= 1<<14,   //启动增强的本地备份功能

= 1<<15,   //启动智能识别加密算法

= 1<<16,   //启用高级加密算法

= 1<<17,   //网络文件采用二代加密算法

= 1<<18,   //启用缓存数据，提高速度

= 1<<19,    //禁止资源管理器预览二代加密文件

= 1<<20,  //在解密模式下，拷贝密文加密

= 1<<21,   //支持微点杀毒软件

= 1<<22,   //网络文件支持多次加密

= 1<<23,    //保存网络文件不是用分页IO

= 1<<24,  //二代加密的文件不清除缓存

= 1<<25,  //允许批量修改文件密级

= 1<<26,  //允许远程桌面拷贝本地文件

= 1<<27,  //仅配置二代加密的程序才进行二代加密

= 1<<28,  //禁用Linux/Mac非受控进程管道传输

= 1<<29,  //不处理驱动上报的共享上的文件加解密请求

例：

EncryptType = 3,ForbitScreen=1,OfflineTime=72小时+0分钟<259200秒>,Open=1,Copy=1,ICON=1,DecryptT=1,CtrlCode=536871040

//以上日志，EncryptType行的CtrlCode是536871040，二进制是0010 0000 0000 0000 0000 0000 1000 0000

从右到左（0开始算）是7、29是1，根据上表，说明“禁止受控程序改名”和“不处理驱动上报的共享上的文件加解密请求”是打钩的





## 12.2 应用程序限制日志——现象日志

ArWorkstation.log

### 12.2.1 应用程序限制

打开系统自带的画图工具，看应用程序限制策略：

**关键字：Terminate**

TerminateProcess Pid = 20084,ExeFile =mspaint.exe,<0>

以上说明设置了应用程序限制，只允许部分程序运行，由于mspaint.exe不在允许范围内，所以被禁止了

### 12.2.2 窗口标题限制

打开文档标题包含有“敏感内容”的word文档，看窗口标题限制策略：

**关键字：Terminate**

**关键字：TitleArrived IsBlock!**

TitleArrived IsBlock!<敏感内容.docx - Word> <敏感内容>

<敏感内容.docx - Word> //表示当前的窗口标题为“敏感内容.docx - Word”

<敏感内容> //表示匹配到了“敏感内容”这个窗口标题限制

### 12.2.3 禁止受控程序改名

关键字：Terminate

**关键字：CheckON222或CheckON111**

CheckON222 para=<PPT.EXE> <POWERPNT.EXE> db=<POWERPNT.EXE> <POWERPNT.EXE>

TerminateProcess Pid = 19200,ExeFile =PPT.EXE,<0>

Insert Process Success! <10a78f80>Pid = 019200 ExeName = PPT.EXE ExePath = C:\Program Files (x86)\Microsoft Office\root\Office16\PPT.EXE,Original = POWERPNT.EXE

表示将受控程序名POWERPNT.EXE改名为非受控程序PPT.EXE，被禁止启动

CheckON222 //表示将可信进程名改成不可信（即：受控程序名改为非受控程序名）

CheckON111 //表示将不可信进程名改成可信（即：非受控程序名改为受控程序名）

db=<POWERPNT.EXE> <POWERPNT.EXE> //表示该程序在受控程序里的设置是文件名和原始文件名都是POWERPNT.EXE

Original = POWERPNT.EXE //表示该程序的原始文件名是POWERPNT.EXE

# 13 USB存储设备限制日志

## 13.1 USB存储设备限制日志——配置日志

ArWorkstation.log

### 13.1.1 USB存储设备限制

关键字DriverBlock或DriverType=<3>

例：DriverBlock Group=<65535> Computer=<0> DriverType=<3> CtrlCode=<1> TimeId=<1>

DriverType=<3>表示这是U盘的规则，CtrlCode , 0=允许，1禁止，2只读，3断网使用

**注：6.0版本在上级分组做了禁止（包括禁止、只读、断网使用）的情况下，可以单独对终端再做允许或其他策略，此时以终端策略为准，日志例如：**

(2020-06-28 17:55:12)[1820][ArWs]DriverBlock Group=<65535> Computer=<0> DriverType=<3> CtrlCode=<1> TimeId=<1> AllowExcept=<0>

(2020-06-28 17:55:12)[1820][ArWs]DriverBlock Group=<0> Computer=<11539> DriverType=<3> CtrlCode=<0> TimeId=<1> AllowExcept=<0>

### 13.1.2 USB存储设备认证

**关键字：GetUsbConfigDb**

例：GetUsbConfigDb GroupId = 0,ComputerId = 11539,DeviceID = 5C070B6A16D7B52B27

表示11539这台终端允许了ID为5C070B6A16D7B52B27的USB存储设备

## 13.2 USB存储设备限制日志——现象日志

ArWorkstation.log

### 13.2.1 插入U盘后，看USB策略

SendParaToDriver Succeed.Type=<1> value=<0> 允许

SendParaToDriver Succeed.Type=<1> value=<1> 禁止

SendParaToDriver Succeed.Type=<1> value=<2> 只读

SendParaToDriver Succeed.Type=<1> value=<3> 断网使用

### 13.2.2 插入USB存储设备后，看设备ID

**关键字：CreateUSBDisk**

例：[ArWs]CreateUsbDisk Device = E:,VolumeName = 可移动磁盘 (E:),DeviceID = 5C070B6A16D7B52B27

**综合：**查询DriverType=<3>看到USB存储设备是禁止、只读或断网使用，插入U盘时查询CreateUsbDisk看到的DeviceID和GetUsbConfigDb查到的所有DeviceID相比对，如果在GetUsbConfigDb的DeviceID列表里，则说明该设备是经过认证且允许这台终端使用的，不受禁止、只读或断网使用的影响。

# 14 行为审计系统

## 14.1 文件监视过滤配置

ArWorstation.log

文件操作对象过滤选项

**关键字：SendFileFilterToDriver process**

**日志示例：**

(2020-07-29 08:45:29)[3996][ArWs]==>SendFileFilterToDriver process=<explorer.exe|winword.exe|notepad.exe|userinit.exe|excel.exe|powerpnt.exe|acrobat.exe|acad.exe|wordpad.exe|client99se.exe|> ext=<2><.doc|.docx|.xls|.xlsx|.txt|.ppt|.pptx|.pdf|.dwg|.ddb|.pcb|>

SendFileFilterToDriver process=<explorer.exe|winword.exe|notepad.exe|userinit.exe|excel.exe|powerpnt.exe|acrobat.exe|acad.exe|wordpad.exe|client99se.exe|> //配置explorer.exe、winword.exe等进程操作的文件都受到监视

ext=<2><.doc|.docx|.xls|.xlsx|.txt|.ppt|.pptx|.pdf|.dwg|.ddb|.pcb|> //配置.doc、.docx、.xls等后缀作为文件过滤后缀选项，参数说明：

ext=<1>【不监视如下后缀的选项】

ext=<2>【只监视如下后缀的选项】

文件备份选项，关键字OPCode

日志示例：

(2020-07-29 09:21:29)[5184]CBackUpOpFile::SetConfig ModuleOpen<0> OpCode<0> MaxSize<0>

……

(2020-07-31 12:00:09)[14636]CBackUpOpFile::SetConfig ModuleOpen<1> OpCode<63> MaxSize<9999>

……

ModuleOpen<0> //文件备份选项是否开启，参数说明：

<0>【不备份文件】

<1>【备份如下操作文件】

OpCode<63> //文件备份开启哪些操作，参数说明：(可以用16进制转2进制，查看对应的勾选内容)

OpCode<1> 【创建文件】

OpCode<2> 【重命名文件】

OpCode<4> 【删除文件】

OpCode<8> 【复制文件】

OpCode<16> 【打开文件】

OpCode<32> 【编辑文件】

OpCode<63>【备份了所有操作】//创建文件、重命名文件、删除文件、复制文件、打开文件、编辑文件

MaxSize<9999>【超过XX M的文件不备份，默认是9999M】//即控制台填写的是0，代表不限制

## 14.2 驱动日志

ArDriver.log

**关键字FileLogList\_SetFilter\_Process: proces=**

日志示例：

(2020-07-29 09:21:41)[5184]0000005184FileLogList\_SetFilter\_Process: proces=|explorer.exe|winword.exe|notepad.exe|userinit.exe|excel.exe|powerpnt.exe|acrobat.exe|acad.exe|wordpad.exe|client99se.exe|

(2020-07-29 09:21:41)[5184]0000005184FileLogList\_SetFilter\_FileExt: code=2, fileext=.doc|.docx|.xls|.xlsx|.txt|.ppt|.pptx|.pdf|.dwg|.ddb|.pcb|

FileLogList\_SetFilter\_Process: proces=|explorer.exe|winword.exe|notepad.exe|userinit.exe|excel.exe|powerpnt.exe|acrobat.exe|acad.exe|wordpad.exe|client99se.exe| //文件监视过滤策略生效，explorer.exe、winword.exe等进程操作的文件都受到监视

FileLogList\_SetFilter\_FileExt: code=2, fileext=.doc|.docx|.xls|.xlsx|.txt|.ppt|.pptx|.pdf|.dwg|.ddb|.pcb| //文件后缀过滤选选项生效，.doc、.docx、.xls等后缀作为文件过滤后缀选项，参数说明：

code=1【不监视如下后缀的选项】

code=2【只监视如下后缀的选项】

# 15 其他日志（绿盾版本、分区名、工作模式、模块编号、MTP等）

## 15.1 绿盾版本

ArWorkstation.log

**关键字Send 0106**

[ArWs]Send 0106 <5> <6.00.200720.SC> <DESKTOP-UDU1HUP> MCode=<8168c78dcf9e748128f7aa3ec606a4d6><10410>

<6.00.200720.SC> //终端版本

<DESKTOP-UDU1HUP> //计算机名

MCode=<8168c78dcf9e748128f7aa3ec606a4d6> //MCOde，终端的唯一标识符GUID相关，不常用

<10410> //终端编号

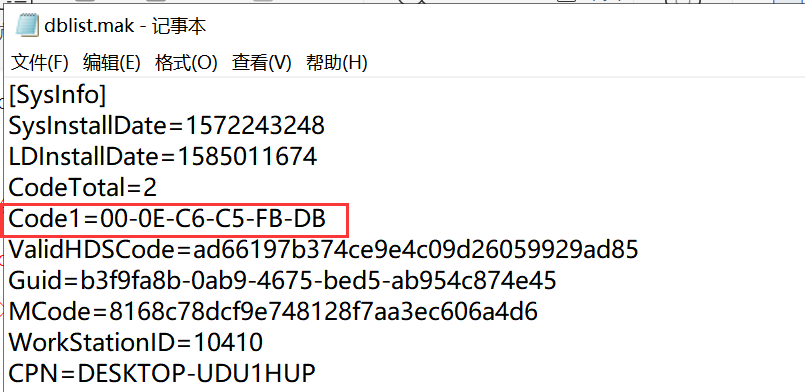
## 15.2 网卡MAC

ArWorkstation.log

**关键字 pAdapt**

pAdapt Address [00-OE-C6-C5-FB-DB]Intel(R) Ethernet Connection (7) I219-V

DbList.mak内容（终端标识文件，位于C:\Windows\System32或C:\Windows\SysWOW64,以及除系统盘外的第一个分区根目录的$Eis$Bak\下，如D:$Eis$Bak\）



**注：如有多张网卡，则日志可以查到多个pAdapt Address，DbList.mak也会有Code2、Code3等**

## 15.3 分区名

ArWorkstation.log

**关键字：m\_DiskVolume**

(2020-08-12 09:15:04)[4452]==>m\_DiskVolume <9672>

(2020-08-12 09:15:04)[4452]AddDevice <C:> <3> <\Device\HarddiskVolume1>

(2020-08-12 09:15:04)[4452]AddDevice <C:> <3> <\Device\HarddiskVolumeShadowCopy1>

(2020-08-12 09:15:04)[4452]AddDevice <D:> <3> <\Device\HarddiskVolume2>

(2020-08-12 09:15:04)[4452]AddDevice <D:> <3> <\Device\HarddiskVolumeShadowCopy2>

(2020-08-12 09:15:04)[4452]AddDevice <E:> <3> <\Device\HarddiskVolume3>

(2020-08-12 09:15:04)[4452]AddDevice <E:> <3> <\Device\HarddiskVolumeShadowCopy3>

(2020-08-12 09:15:04)[4452]AddDevice <F:> <3> <\Device\HarddiskVolume4>

(2020-08-12 09:15:04)[4452]AddDevice <F:> <3> <\Device\HarddiskVolumeShadowCopy4>

(2020-08-12 09:15:04)[4452]AddDevice <G:> <3> <\Device\HarddiskVolume5>

(2020-08-12 09:15:04)[4452]AddDevice <G:> <3> <\Device\HarddiskVolumeShadowCopy5>

(2020-08-12 09:15:04)[24516]AddDevice <H:> <2> <\Device\HarddiskVolume6>

(2020-08-12 09:15:04)[24516]AddDevice <H:> <2> <\Device\HarddiskVolumeShadowCopy6>

(2020-08-12 09:15:04)[4452]AddDevice <\\> <4> <\Device\LanmanRedirector\>

(2020-08-12 09:15:04)[4452]AddDevice <\\> <4> <\??\UNC\HostName\Share\File\>

(2020-08-12 09:15:04)[4452]AddDevice <\\> <4> <\UNC\>

(2020-08-12 09:15:04)[4452]AddDevice <\\> <4> <UNC\>

(2020-08-12 09:15:04)[4452]AddDevice <\\> <4> <\Device\Mup\>

AddDevice <C:> <3> <\Device\HarddiskVolume1> //表示C盘的分区名是\Device\HarddiskVolume1

**注：**

（1）由于三代加密驱动识别文件路径时，路径名不是以类似C:\开头的，而是以底层驱动路径的分区名开头，所以在做特殊目录设置、受控程序高级设置时，如果涉及到盘符，不能用C:\这样的，而要到ArWorkstation.log确认该盘符对应的分区名是什么，再进行设置，比如\Device\HarddiskVolume1

（2）以上示例日志中的<2>、<3>、<4>，是指驱动器类型，<1>代表软盘（已淘汰）；<2>代表USB可移动磁盘；<3>代表本地硬盘；<4>代表网络驱动器（\Device\Mup\、\UNC\、<\Device\LanmanRedirector\都是网络路径名，如果特殊目录等设置要针对网络路径，可通过ArDriver.log或ArWorkstation.log查看具体是哪种，常见的是\Device\Mup\，比如要设置[\\192.168.1.5](file:///\\\\192.168.1.5)为特殊目录，则应该设置为\Device\Mup\192.168.1.5\，直接设置为\192.168.1.5\也可以，但更容易被冒充）

## 15.4 时间规则

ArWorkstation.log

**关键字：GetTimeInfoFromDB**

(2020-08-12 09:15:10)[4452]GetTimeInfoFromDB <1> <067d6c90>

(2020-08-12 09:15:10)[4452]<00:00:00-23:59:59>

(2020-08-12 09:15:10)[4452]<00:00:00-23:59:59>

(2020-08-12 09:15:10)[4452] <00:00:00-23:59:59>

(2020-08-12 09:15:10)[4452] <00:00:00-23:59:59>

(2020-08-12 09:15:10)[4452] <00:00:00-23:59:59>

(2020-08-12 09:15:10)[4452] <00:00:00-23:59:59>

(2020-08-12 09:15:10)[4452] <00:00:00-23:59:59>

//<1>表示时间规则1，一个时间规则会有7条信息，从上到下分别是星期一到星期天的时间定义，时间规则在设置桌管策略、设置工作模式时可能会用到，曾经碰到过USB存储设备限制不生效，发现是时间规则定义不对的原因。

## 15.5 工作模式

Arworkstation.log

### 15.5.1 指令策略更新

**关键字：WorkModeSwitch**

(2020-08-10 00:24:48)[8872][LdWs]Download file succeed.<WorkModeSwitch> UsedTime=<78> FileSize=<28>B

(2020-08-10 00:24:48)[8872][ArWs] CProtectFiles::UpdateFile1(WorkModeSwitch.db)!

(2020-08-10 00:24:48)[8872][ArWs]SaveModifyTime.<WorkModeSwitch> <2020-08-10 00:22:00> <2020-08-10 00:24:47>

(2020-08-10 00:24:48)[8872]SetWorkType Succeed.Type New=<1>,Old=<1>

(2020-08-10 00:24:48)[8872][ArWs]ModuleFilter:Select ModuleID From ModuleFilter where ( GroupId = 65535 ) or ( GroupId = 1 ) or ( ComputerId = 10412 )

UpdateFile1(WorkModeSwitch.db)：指令成功更新到数据库WorkModeSwitch.db中的

### 15.5.2 策略生效

ArWorkstation.log

**关键字 WorkMode**

##### 15.5.2.1 工作模式启用加密

[ArWs] GetTermWorkModeFromDB WorkMode. TermHasWorkMode = 1, WorkModeCmd = 0,ReadDbInfo = 1

或

[ArWs] DealMMFrame0408 WorkMode. TermHasWorkMode = 1, WorkModeCmd = 0,AnswerType = 1

TermHasWorkMode = 1 //0表示禁止切换工作模式，1表示允许切换工作模式

WorkModeCmd = 0 //当前工作模式状态为启用加密

##### 15.5.2.2 工作模式关闭加密

[ArWs] DealMMFrame0408 WorkMode. TermHasWorkMode = 1, WorkModeCmd = 1,AnswerType = 1

WorkModeCmd = 1 //当前工作模式状态为关闭加密

TermConfig Driver Can not Work, Type = 9 //驱动停止工作，工作模式切换为不启动

SetWorkType Succeed.Type New=<5>,Old=<5> //进入模式为5的状态，即注销状态

[LdWs]CFileEncrypt CheckToCloseProcessByWorkMode!<outlook.exe> <0> //工作模式为关闭加密时，禁止outlook.exe

##### 15.5.2.3 附：驱动停止工作代码byCantWorkCause

1 企业密钥个数为0

2 termid为0

3 typeid为0

4 终端不能工作

6 服务端断开太久

7 没有USBKEY

9 工作模式切换为不启动

10 主密钥异常

此外，工作模式为关闭加密时，ArDriver.log会有以下日志：

FileEncryptService\_04: cmd=5, NeedEncrypt=1

cmd=5 //表示注销状态（工作模式关闭加密或终端未登录）

NeedEncrypt=1 //对文件进行加密，实际上值是由Cmd值决定的，不会理会这个NeedEncrypt=1

附：Cmd各个值表示：1启动、2停止、5注销、6只解密不加密、7外发终端停止

## 15.6 查看功能模块是否启用

ArWorkstation.log

**关键字：GetModuleInfo或Module is Using**

[ArWs]GetModuleInfo Succeed.<29>

[ArWs]Module is Filter!<73> 模块被过滤

[ArWs]Module is Using! <73> 模块被使用

[ArWs]Module is not Using!<73> 模块没有分配

### 15.6.1 模块编号列表

|  |  |  |
| --- | --- | --- |
| **模块名** | **编号** | **含义** |
| enModuleApp | 1 | 应用程序日志 |
| enModuleFileOp | 2 | 文件操作日志(CDocOpReport InsertDocOp) |
| enModuleScreenRealTime | 3 | 屏幕实时监视日志 |
| enModuleScreenReplay | 4 | 屏幕回放日志 |
| enModulePrinter | 5 | 打印机日志 |
| enModuleHttp | 6 | Http日志 |
| enModuleMail | 7 | 邮件日志 |
| enModuleFtp | 8 | Ftp日志 |
| enModuleWindowTitle | 9 | 窗口标题 |
| enModuleDev | 10 | 设备模块 |
| enModuleComputerCtrl | 11 | 计算机控制模块 |
| enModuleChatCapture | 12 | 聊天监控模块 |
| enModuleFileEncrypt | 20 | 文件加密模块 |
| enModuleArpWall | 21 | Arp防火墙模块 |
| enModuleRemoteTaskMgr | 22 | 远程任务管理器 |
| enModuleAlarm | 23 | 报警 |
| enModuleAssetMgr | 24 | 资产管理 |
| enModuleBackupFile | 25 | 文件备份模块 |
| enModuleService | 26 | 服务监视模块 |
| enModuleInternetCtrl | 27 | 上网限制 |
| enModuleRealLog | 28 | 实时日志 |
| enModuleAppAnalysis | 29 | 应用程序分析模块 |
| enModuleWinUser | 30 | windows操作员监视 |
| enModuleAppBlock | 31 | 应用程序限制 |
| enModuleHardwareBlock | 32 | 硬件限制 |
| enModuleOSF | 33 | 文件直接外发 |
| enModulePrintMonitor | 34 | 打印内容监控模块 |
| enModuleAppStat | 35 | 不统计应用程序信息 |
| enModuleNetFlow | 36 | 流量控制 |
| enModuleSendMail | 37 | 发送邮件模块 |
| enModuleIPMACBind | 38 | IP-MAC绑定模块 |
| enModuleUsbIdentify | 39 | USB认证 |
| enMoudleFilesManage | 40 | 文件集中管理 |
| enMoudleLdxc | 41 | 内部流转 |
| enMoudleWebMail | 42 | 网页邮件监控 |
| enMoudleUpLoadFile | 43 | 网页上传文件监控 |
| enMoudleFileOp2 | 44 | 文件操作日志（与2同）20110830 |
| enMouuleQQFileSendMonitor | 45 | QQ传输文件监控 |
| enModuleHttpServer | 46 | 服务端白名单模块 |
| enMoudleQQProtect | 47 | QQ防护模块 |
| enMoudleNewMailWhiteList | 48 | 新邮件白名单 |
| enMoudleApplyPrint | 49 | 打印申请及审批 |
| enModuleNetRestrict | 50 | 关闭流量限制 |
| enModuleComputerBaseInfo | 51 | 终端基本信息 |
| enModuleSoftManage | 52 | 软件管理 |
| enModuleDeviceManage | 53 | 设备管理 |
| enMoudleShareManage | 54 | 共享管理 |
| enModuleResManage | 55 | 资源管理 |
| enModuleSoftHardwareLog | 56 | 软硬件安装日志 |
| enModuleTaskPush | 57 | 任务推送 |
| enModuleFileOpAudit | 58 | 文件操作日志\_审计系统 |
| enModulePrintMonitorAudit | 59 | 打印内容监控模块\_审计系统 |
| enModuleScreenWater | 60 | 屏幕水印 |
| enModuleCD\_burning | 61 | 光盘刻录 |
| enModuleNetSegregate | 62 | 网络隔离模块 |
| enModuleDiskEncryptLog | 64 | 全盘加解密日志 |
| enModulePrintWaterMark | 68 | 打印水印模块 |
| enModuleQQFileUp | 69 | 聊天工具文件监控 |
| enModuleAppInfoGather | 70 | 应用程序收集 |
| enModulePatchManagement | 71 | 补丁管理 |
| enModuleUsbOfflineUse | 72 | USB存储设备断网使用 |
| enModuleInsallPackBlock | 73 | 软件安装包限制 |
| *enModuleDailyUrl* | *101* |  |
| *enModuleDailyMail* | *102* |  |
| *enModuleDailyTalk* | *103* |  |
| *enModuleDailyFtp* | *104* |  |
| *enModuleMsnChat* | *105* |  |

## 15.7 操作员名称、操作员编号、分组编号、文件密级等

ArWorkstation.log

**关键字 UserName**

[ArWs]TermUser TermId<121> TypeId<30> UserName<User19955> UsbOnline=<0> ComputerId=<0> GroupId=<136> Level=<2>

//当前登录的操作员名称是User19955,操作员编号是121，分组编号是136，操作员类型编号是30，文件密级是2（即“秘密文件”），ComputerId=<0>表示该终端操作员未绑定任何终端，如有绑定，则会显示绑定的终端编号

### 15.7.1 附：默认的文件密级对应关系

0 公开文件

1 内部资料文件

2 秘密文件

3 机密文件

4 绝密文件

### 15.7.2 允许或禁止MTP

ArDriver.log

关键字 FileEncryptService\_28: type=11

FileEncryptService\_28: type=11, value=1 //说明当前的设置是允许手机MTP拷贝，vaule =1 允许 vaule=0 禁止

**受到MTP限制的现象日志：**

**关键字 Forbit Phone**

[setup.exe]Forbit Phone FileName<\Device\HarddiskVolume2\桌面\install\driver\gdi\amd64\WinUSBCoInstaller.dll>

//安装程序生成WinUSBCoInstaller.dll时被MTP禁止了（导致安装失败）

**注：禁止MTP引起的问题一般是手机MTP拷贝不能用、苹果手机不能充电、笔记本电脑指纹不能用、某些USB设备（如烧录设备）不能用、某些软件安装无法安装等**

### 15.7.3附：禁止MTP所禁止的文件列表

wudfrd.sys

winusb.sys

usbaapl.sys

winusbcoinstaller2.dll

winusbcoinstaller.dll

usbaapl64.sys

wpdusb.sys

# 16 错误码与ini

## 16.1 常见错误码

### 16.1.1 注册失败错误码

0 //表示成功，其他表示失败

1~99 //一般错误

100~199 //联网验证的失败码

200~299 //为文件注册的失败码

1 //参数错误

2 //序列号非法

3 //序列号类型错误

4 //获取注册帧失败 （一般不可能产生）

110 //升级过期

111 //软件狗不对

112 //注册次数超出

113 //试用过期

114 //产品类型不同

115 //未找到该用户记录

116 //注册网站忙（实际是网站数据库连接不上）

117 //查询错误

150 //协议格式错误

151 //传输错误（中途信息被修改）

152 //非法修改主密钥

153 //试用时间已到

201 //文件格式错误

202 //文件版本错误

203 //文件长度错误

204 //文件校验错误

205 //文件内容检验错误（加密内容的校验）

206 //文件未绑定硬件信息

207 //绑定硬件信息错误

208 //软件版本错误（软件版本日期不在允许日期内）

209 //试用天数错误

210 //非法更改主密钥

211 //打开注册文件错误

212 //创建序列号失败 （一般不会发生）

213 //文件中的产品类型错误

214 //试用已过期

215 //文件转换后读取失败 //新增加

253 //查询注册信息失败

254 //网站应答的格式错误

255 //连接网站失败

12007 //表示服务器名称无法被解析（ERROR\_INTERNET\_NAME\_NOT\_RESOLVED）。也就是说，这个HTTP请求根本没发到服务器上,一般是DNS设置的问题。

12029 //一般是https的方式连接不上，可以通过修改ArServer.ini改为用http方式连接，详见3.2.4

### 16.1.2 终端操作员登录失败错误码

1 //操作员不存在或密码错误

2 //usb key不正确

4 //绑定的终端不对

5 //禁止空密码登录

6 //密码错误

### 16.1.3 驱动停止工作错误码

**终端红色减号，日志搜索关键字Driver Can not Work**

例如：TermConfig Driver Can not Work, Type = 3

can not type=0 //密钥发生异常：原来有密钥，变成没密钥后，停止驱动。（改名2001-01-01）

can not type=1 //企业密钥数量为0  停止工作。红色加号，没有分配模块

can not type=2 //设备没编号，终端类型没有

can not type=3 //没有终端类型，换个终端类型试看看

can not type=6 //超过短期离线时间

can not type=11 //在线时间限制，超过限制时间后就算在线也不能用（控制台的终端设置 - 常规设置 - 终端在指定日期后关闭加解密功能）

### 16.1.4 导入离线策略错误码

-1 //打不开离线文件

-2 //读取离线文件头失败

-3 //检验离线文件头失败，认为不是离线文件

-4 //文件校验失败，认为文件有被非常修改

-5 //密码不对（密码是NULL，这个基本不会）

-7 //超期了

-8 //不是给这个终端的离线文件

-9 //主密钥不匹配（终端服务这边导入离线策略，是在主密钥非空且导入策略的主密钥跟当前不同时会返回-9）

-10 //密码不对（输入的密码是空字符串）

-11 //文件名为空

### 16.1.5 申请解密失败错误码

|  |  |  |
| --- | --- | --- |
| **参数名** | **错误码** | **中文说明** |
| dsDecryptOK | 0 | //成功 |
| dsFileError | 1 | // |
| dsNotEncryptFile | 2 | // |
| dsPwdNotFound | 3 | // |
| dsPwdError | 4 | //终端保存的密钥信息与文件中保存的密钥信息不一致 |
| dsHeaderError | 5 | //一代加密文件头512解压失败 |
| dsNotSupport | 6 | //不支持的加密文件版本，超过当前已知的版本 |
| dsWriteError | 7 | //往加解密的临时文件中写入加解密后的内容失败 |
| dsReadError | 8 | // |
| dsMoveFileError1 | 9 | //是目标文件存在（比如把A文件解密到B文件，则B是目标文件，大多时候我们不指定把文件解密到另一个位置只是要替换原文件，那目标文件也是原文件），需要先将它改名，但改名失败了 |
| dsMoveFileError2 | 10 | //是删除改名后的原文件失败 |
| dsMoveFileError3 | 11 | //是把加解密出来的临时文件改名为目标文件失败 |

此外，申请解密还有一个错误码比较常见，是数据库查询超时<0002>，例如：

[2020-04-02 16:53:32]SqlError:Insert Into ApprovalDecryptLog

(InfoId,FunctionId,ComputerId,TermUserId,GroupId,FlowId,BeginTime,CurrsetpId,SrcPath,SavePath,FileSize,LastModifyTime,ReqMsg,State,Result,FileCode,FileState,BackUpPath,IsDir,RecvDepartment,RecvName,RecvTel)Values('{0D44B444-576F-4020-BFC1-BCC27F473C00}',1,2627,936,142,4,'2020-04-02 16:53:22',1,'D:\潘宇\报价修正\WISH燕邮宝平邮-特货.xls',

'$APPPATH$\DataFile\2020-04-02\ApprovalUpLoad\{0D44B444-576F-4020-BFC1-BCC27F473C00}',30208,'2020-04-02 16:48:32','',0,0,'c1e0cc8a6cf749bfc272d554c3f8aaf8',0,'C:\Inetpub\ftproot\Tipray\LdTerm\Repository\2020-04-02\DataFile\ApprovalUpLoad\{0D44B444-576F-4020-BFC1-BCC27F473C00}',0,'','','' )<0002>

### 16.1.6 压缩包穿透加解密错误码

|  |  |  |
| --- | --- | --- |
| **参数名** | **错误码** | **中文说明** |
| TR\_SUCCEED | 0 | //正常，没有错误 |
| TR\_OPEN\_FILE\_FAIL | 10001 | //打开文件失败 |
| TR\_FILE\_TOO\_SMALL | 10002 | //文件是空的 |
| TR\_FILE\_TYPE\_ERROR | 10004 | //压缩失败 |
| TR\_FILE\_ALREADY\_ENCRYPT | 10005 | //文件已经是加密的 |
| TR\_FILE\_ALREADY\_DECRYPT | 10006 | //文件已经是解密的 |
| TR\_FILE\_ENCRPT\_ERROR | 10007 | //加密过程错误 |
| TR\_CREATE\_FILE\_ERROR | 10008 | //创建文件失败 |
| TR\_CREATE\_ENCRYPT\_HEAD\_FAIL | 10009 | //创建加密头失败 |
| TR\_WRITE\_FILE\_FAIL | 10010 | //写文件失败 |
| TR\_READ\_FILE\_FAIL | 10011 | //读文件失败 |
| TR\_MOVE\_FILE\_FAIL | 10012 | //改名失败 |
| TR\_DELETE\_TEMP\_FILE\_FAIL | 10013 | //删除临时文件失败 |
| TR\_GET\_KEY\_FAIL | 10014 | //密钥错误 |
| TR\_FILE\_DECRYPT\_ERROR | 10015 | //解密过程失败 |
| TR\_FILE\_VERSION\_NOT\_AVAILABLE | 10016 | //该版本尚未支持本操作 |

### 16.1.7 独立审批：终端与独立审批通信返回的值（一般会在LdApprovalEx.log体现）

独立审批是通过http接口去处理，所以错误码是跟网页的错误码有关：

-1: 与审批服务器断开连接！ //返回响应（GET与PUT请求）

200: 操作成功！ //返回响应（GET与PUT请求）

201: 操作成功！ //已经创建了实体，并在响应体中返回（POST请求）

204: 操作成功！ //已经删除了实体，因此没有返回的响应体（DELETE请求）

401: 操作失败！ //操作要求设置认证头。如果请求中有认证头，则提供的鉴证并不合法，或者用户未被授权进行该操作。

403: //操作被禁止，且不应重试。

404: //操作失败！请求的资源未找到。

405: //操作失败！使用的方法不能用于该资源。

409: //操作失败！该操作导致更新一个已被其他操作更新的资源，因此本更新不再有效。

415: //操作失败！请求体包含了不支持的媒体类型。

500: //操作失败！执行操作时发生了未知异常。

### 16.1.8 移动终端错误码（包括登录和处理集成审批请求的）

|  |  |  |
| --- | --- | --- |
| **参数名** | **错误码** | **中文说明** |
| E\_OPT\_SUCCESS | 1 | //成功 |
| E\_OPT\_ERROR | 2 | //执行错误 |
| E\_OPT\_EQUIPMENT\_NOT\_ALLOW | 3 | //设备不允许接入（禁用） |
| E\_OPT\_EQUIPMENT\_WAIT | 4 | //设备等待审核 |
| E\_OPT\_REQUEST\_INFO | 5 | //请求信息错误 |
| E\_OPT\_USER\_NAME | 6 | //用户名错误，或者不存在 |
| E\_OPT\_USER\_PASSWORD | 7 | //密码错误 |
| E\_OPT\_EQUIPMENT\_NOT\_REG | 8 | //设备未注册 |
| E\_OPT\_APPLY\_REVOKE | 9 | //申请已撤销 |
| E\_OPT\_APPLY\_ALREADY\_APPROVAL | 10 | //申请已被审批 |
| E\_OPT\_CANNOT\_DOWNLOAD\_FILE\_SIZE\_TOO\_BIG | 11 | //文件太大不能下载 |
| E\_OPT\_CANNOT\_DOWNLOAD\_FILE\_IS\_FOLDER | 12 | //文件夹不能下载 |
| E\_OPT\_CANNOT\_DOWNLOAD\_NO\_UPLOAD | 13 | //文件未上传不能下载 |
| E\_OPT\_NOT\_MOBILE\_MANAGE | 14 | //没有移动终端管理模块权限 |
| E\_OPT\_NOT\_BUY\_SAFE\_PRODUCT | 15 | //没有购买防泄密产品 |
| E\_OPT\_USER\_OVER\_TIME | 16 | //试用到期 |
| E\_OPT\_DEVICE\_ACCOUNT\_BIND | 17 | //设备已绑定用户 |
| E\_OPT\_APPROVAL\_VERIFY\_CODE\_LENGTH | 18 | //审批申请码长度错误 |
| E\_OPT\_APPROVAL\_VERIFY\_CODE\_FORMAT | 19 | //审批申请码格式错误 |
| E\_OPT\_APPROVAL\_VERIFY\_CODE\_FLOW | 20 | //审批验证码不是自己的流程 |
| E\_OPT\_PORT\_BIND | 21 | //引擎服务连接不上 |
| E\_OPT\_ENGINE\_CONNECT | 1000 | //端口绑定错误 |
| E\_OPT\_REMOTE\_EQUIPMET\_ALLOW | 2000 | //远程查询设备准入成功 |

### 16.1.9 批量加密错误码

1 //无法打开文件，文件不存在

2   //读原始文件失败，也有可能是这个文件内容是空 0字节，或者加密类型可能是“打开文件后加密”

3 //文件夹没有写入权限，可以添加everyone完全控制权限试试

5 //文件已经是密文，无需加密

6 //无法加密，需要开启高级加密功能才行（可能勾选了“仅配置二代加密的程序才进行二代加密”，这种情况下可以配置LdTermDaemon.exe对指定后缀二代加密）

7、8、9 //文件改名失败（如文件被占用），假设是三代加密，批量加密失败代码8，有可能是受控了索引进程（SearchProtocolHost.exe）

101 //无法打开文件，文件存在，其他原因（如文件被占用、路径太长）

### 16.1.10 批量解密错误码

1 //无法打开文件，文件不存在

2 //读原始文件失败，也有可能是这个文件内容是空 0字节，或者加密类型可能是“打开文件后加密”

3 //文件夹没有写入权限，可以添加everyone完全控制权限试试

4 //密钥错误

5 //文件已经是明文，无需解密（特殊案例：若文件确实是密文还提示错误码5，检查是否受控LdTermDaemon.exe进程，或者操作员类型是否“打开文件后加密”的类型）

7、8、9 //文件改名失败（如文件被占用），假设是三代加密，批量加密失败代码8，有可能是受控了索引进程（SearchProtocolHost.exe）

15 //改名或者复制失败

17 //拷贝失败 （可能权限问题或空间不足）

18 //读取异常

100 //密钥错误

101 //无法打开文件，文件存在，其他原因（如文件被占用、路径过长）

112 //磁盘空间不足

242 //密钥错误，密钥序号0，要用工具解密，联系研发

另外，如果result数字非常大 那么也是密钥错误

### 16.1.11 应用接入系统错误码

8193 //打开数据库失败

8194 //接收的是空内容

8195 //等待审批

8196 //审批不通过

8197 //密码错误（用户登入或修改密码时）

8198 //数据库操作失败，插入，修改，查询

**8199 //在准入设备表里查询不到此设备**

8200 //注册码无效

8201 //数据不完整

8202 //没有应用系统

8203 //其他错误

8204 //账号禁用

8205 //账号未绑定此设备

8206 //审批不通过

8207 //此账号不存在

8208 //此终端不用升级

8209 //系统未注册

8210 //系统终端数量超过

8211 //注册时间超过

8212 //狗不匹配

8213 //没有狗的信息

8214 //此用户没有绑定狗

12289 //未登录

12290 //未知错误

12291 //socket错误

12292 //找不到dll

## 16.2 常用ini

### 16.2.1 LdDataServer.ini

linux和macOS终端要正常下发策略，需要在LdDataServer.ini里加上以下两行：

[SettingsLinuxMac]

CanCreateTactics=1

### 16.2.2 LdConsole.ini

（1）要用其他端口登录控制台（比如用花生壳内网穿透的），需要在LdConsole.ini的LoginIp行之下增加一行：

LoginPort=15083 //比如端口号是15083

**注：**控制台一般是老板或网管使用，如果有外网登录的情况，有可能是笔记本有时要在公司内网登录，有时要在外网登录，但是端口号一旦设置，登录内网时也会用这个端口，就登录不上了。解决办法是控制台默认安装目录下的LdConsole.ini不改，另外复制一个控制台目录到其他位置，在这个目录下的LdConsole.ini里改端口，然后再创建个快捷方式到桌面，两个快捷方式可以标注一个内网一个外网。

（2）离线策略文件里选择的时间范围，默认最长是180天，如果要设置更长的，可以在LdConsole.ini里增加两行：

[Param]

OfflineDateLimit=366 //比如最长要允许设置366天

（3）开启禁止删除加密文件功能。5.55以上版本默认取消了禁止删除加密文件功能，要使用该功能，可以在LdConsole.ini的[Param]之下增加一行：

ForbidDelEnc=1

（4）在开启了“允许通过资源管理器下载终端文件”的情况下，拷贝文件和文件夹，默认有大小上限，文件大小上限30M，文件夹大小上限300M。如果要更改，是在LdConsole.ini里加上这个：

[FileSetup]

MaxFileSize = 30

MaxFolderSize = 300

单位是MB，根据需要调整大小

### 16.2.3 User.ini

在某些情况下，服务器获取到不同终端的MAC地址信息相同可能会导致串号（比如VPN、虚拟机之类的虚拟网卡相同），可以在引擎服务程序目录下新建一个User.ini文件以解决此问题。例如控制台显示的问题终端MAC地址是00-50-56-c0-00-01，则User.ini内容如下：

[MAC]

Count=1

Mac1=005056c00001

意思是获取终端MAC地址时，如果碰到这个MAC则跳过，获取其他MAC，这样就可以避免上述问题。如果串号的MAC不止一个，则Count改成相应的个数，例如：

[MAC]

Count=2

Mac1=005056c00001

Mac2=005056c00008

### 16.2.4 ArServer.ini

**（1）SSLType=0**

绿盾联网注册方式默认是走https协议，即后台访问https://www.tipray.com有的操作系统支持不了我们的ssl证书所用的算法（SHA256），常见于2003系统。这种情况下可以改成使用http协议，可在ArServer.ini里加上两行：

[System]

SSLType=0

注：也可以通过打KB968730补丁的方式让2003系统支持SHA256算法，这样不用改成http也能正常注册。

**（2）OnlyUseMac=1**

对于虚拟桌面（或叫云桌面，常见的有思杰、VM）、无盘系统，不能用DbList.mak来识别终端，否则会导致同一个虚拟桌面的用户都是同一个终端编号。这种情况下首先要打钩终端高级设置里的“不保存终端标记（如云桌面母盘的终端）”，并且要在引擎服务程序的ArServer.ini里加上以下两行，即改成只用MAC地址来识别终端：

[ServerSet]

OnlyUseMac=1

### 16.2.5 ArVsCfg.ini

用于给引擎或采集设置定时重启。大客户终端数量多、数据量大，服务器运行时间长了可能引擎或采集占用CPU、内存资源会很高，可以设置引擎或采集定时重启进程、释放资源。方法是把服务器目录下的ArVsCfg.ini拷贝到引擎或采集目录下，ArVsCfg.ini填写类似以下内容：

[TimeCtrl]

#时间必须格式化成 00:00

Count = 1

Time1 = 02:10

//表示每天凌晨02:10自动重启引擎或采集（放在引擎目录是重启LdServer.exe，放在采集目录是重启LdData.exe）

**注：**如果一天要重启多次，则Count改成相应数值，再加上Time2、Time3…例如：

[TimeCtrl]

#时间必须格式化成 00:00

Count = 2

Time1 = 02:10

Time2 = 12:10

### 16.2.6 ArComm.ini

终端的ArComm.ini，可以查看本终端连接的服务器地址、端口、数据采集服务器ID，也可修改。用于技术人员不知道控制台密码的情况下修改服务器地址端口，**不能告诉客户此方法。**

[CommConfig]

PsId=1

HasGateway=1

CommServerIp=192.168.1.2 //主服务器地址

CommServerPort=20081 //主服务器端口

CommIp\_Main=117.25.167.254 //辅助服务器地址

CommPort\_Main=20081 //辅助服务器端口

GatewayId=33 //数据采集服务器ID

### 16.2.7 Fotbit.ini

直接外发文件的窗口标题关键字限制信息。直接外发文件的另存、导出、输出等等操作默认是被禁止的，通过终端目录下的Forbit.ini来定义。如果发现外发文件的某种操作没有被禁止，导致可以保存出明文，可以在Forbit.ini里增加那个操作的窗口标题关键字（Count值也要相应增加1）；或者外发文件必须用到某个操作，但又被禁止了，可以看下Forbit.ini里是否有这个关键字，删除掉（Count值也要相应减少1，但由于这个关键字下面的每一条的编号也要修改，不方便，通常是直接把这个关键字改成其他无关字符，Count值也不用改了），例如：

[Key]

Count=161

Key1=另存

Key2=SAVE AS

Key3=导出

…… //太多了，省略

Key160=发到QQ空间

Key161=登录新浪微博

如果要增加一个限制关键字Weibo Login，就是：

[Key]

Count=162

Key1=另存

Key2=SAVE AS

Key3=导出

…… //太多了，省略

Key161=登录新浪微博

Key162=Weibo Login

### 16.2.8 ArsDbClient.ini

可以修改数据库查询、写入超时阈值，以解决因超时引起的终端申请解密数据保存失败等问题，默认10秒超时，也就是说申请解密查询或写入数据库超过10秒就认为超时，会提示“申请解密失败，数据保存失败”。可以在终端目录下新建ArDbClient.ini内容如下：

[System]

WaitTime = 30 //数据库超时阈值调整为30秒

### 16.2.9 WinTrSys.ini

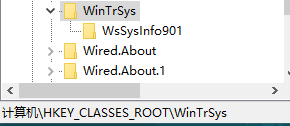
清除注册信息时要删除的文件之一。有时会碰到客户已经注册了，但注册信息有问题，比如控制台登录进去一直是空白的，再重新注册也一直提示注册失败，此时就要清空注册信息再重新注册了，绿盾清除注册信息的方法是：

**绿盾清空注册信息：**

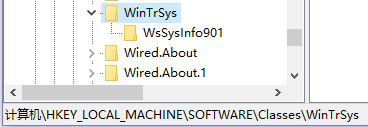
删除以下注册表和文件，然后重启引擎后重启控制台再注册：

（1）查找注册表项，找到所有WinTrSys\WsSysInfo901，删除。一般是以下两个注册表项：

\HKEY\_CLASSES\_ROOT\WinTrSys\WsSysInfo901



\HKEY\_LOCAL\_MACHINE\SOFTWARE\Classes\WinTrSys\WsSysInfo901



（2）查找系统盘下的C:\Windows\SysWOW64\WinTrSys.ini文件，删除。

（3）删除引擎目录下的注册文件信息

trInfoPf1.irs

trInfoPf2.irs

（4）如果是文件注册，还会有leaderPfv1.lic，也要删除

### 16.2.10 ArReRights.ini（详见《关于绿盾特殊功能开启授权说明（仅内部使用）.pdf》）

（1）这是特殊功能的授权文件，销管开通特殊功能后，会给一个ArReRights.ini，放到控制台指定目录下即可；

（2）其中，需要放到控制台目录（如D:\Program Files\Tipray\LeaderEIS\控制台\）的功能有：允许通过资源管理器下载终端文件、控制台远程协助终端时不用对方同意、允许解密竞品（包括无缝替换竞品、绿盾一代转二代以及绿盾更换主密钥）、允许使用半透明加密、允许生成加解密接口授权文件、允许生成重置控制台登录密码授权文件；

（3）需要放到引擎服务程序目录（如D:\Program Files\Tipray\LeaderEIS\服务器\引擎服务程序\）的功能有：离线终端不需要联网验证；

（4）需要同时放到控制台目录、引擎服务程序目录、移动终端服务器安装目录（如D:\Program Files\Tipray\TRMP\）的功能有：允许移动终端使用Windows终端的模块（即可以动态调整移动终端和Windows终端的模块）；

（5）不需要授权文件，只要开通后服务器重新注册即可的有：启用敏感内容识别（即智能加密）、企业密钥。

### 16.2.11 Debug.ini（谨慎使用）

大客户终端数量多、数据量大，有些情况下可能会出现终端大面积掉线等问题，此时可以用Debug.ini彻底关闭服务器的一些功能，看看到底问题出在哪个功能上。Debug.ini放在数据采集服务器目录下，文件内容如下（1表示开启此功能，0表示关闭此功能，编号和功能对应列表在文件里有说明）：

[DebugSetup]

Count = 17

1 = 1

2 = 1

3 = 1

4 = 1

5 = 1

6 = 1

7 = 1

8 = 1

9 = 0

10 = 0

11 = 0

12 = 0

13 = 0

14 = 0

15 = 0

16 = 0

17 = 0

miDebugCheckToCollect = 1, //收集日志

miDebugCheckToAppStat = 2, // 收集工作站应用程序统计信息

miDebugCheckToCollectDev = 3, // 收集工作站软硬件配置

miDebugCheckToCollectScreen = 4, // 收集屏幕数据

miDebugCheckToBackupFile = 5, //备份文件

miDebugCheckToCollectNetFlux = 6, //收集流量统计信息

miDebugMdlFileBackup = 7, //文件自动备份

miDebugMdlFileDecrypt = 8, //解密文件

miDebugMdlFileEMail = 9, //EMail文件

miDebugMdlFileOutSend = 10, //外发文件

miDebugMdlFilePrinter = 11, //打印监控文件

miDebugMdlFileBatchEncrypt = 12, //批量加密日志

miDebugMdlFileBatchDecrypt = 13, //批量解密日志

miDebugCheckToApprovalProcess = 14,//审批流程

miDebugMdlFileUpLoad = 15,//文件网络上传 //网络文件上传guoyx

miDebugMdlApprovalUpLoad = 16,//审批文件上传 guoyx 20110304

miDebugMdlQQFileSendMonitor = 17,//qq文件上传 guoyx 20111129

[DebugSetup]

NeedData = 1

NeedDataTime = 1000